Face perception in videos : contributions to a visual saliency model and its implementation on GPUs

Anis Ur Rahman

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Nano-Electronique et Nano-Technologies**

Arrêté ministériel : 7 août 2006

Présentée par

Anis ur Rahman

Thèse dirigée par **Dominique Houzet et Denis Pellerin**

préparée au sein du **GIPSA-lab**
et de l'École Doctorale **Electronique, Electrotechnique, Automatique & Traitement du Signal**

Face perception in videos: Contributions to a visual saliency model and its implementation on GPUs

Thèse soutenue publiquement le **12 April, 2013**
devant le jury composé de:

Mr. Alain Tremeau
Université Jean Monnet, Saint-Etienne, France, Président
Mr. Simon Thorpe
Research Director at CNRS Toulouse, France, Rapporteur
Mr. Christopher Peters
KTH Royal Institute of Technology, Sweden, Rapporteur
Mr. Michel Paindavoine
Université de Bourgogne, France, Examinateur
Mr. Dominique Houzet
Institut Polytechnique de Grenoble, Grenoble, France, Directeur de thèse
Mr. Denis Pellerin
Université Joseph Fourier, France, Co-Directeur de thèse

To the memory of my uncle, Dr. Niamatullah Khan, whose passion for science inspired me in my graduate studies and in life.

Abstract

Studies conducted in this thesis focuses on faces and visual attention. We are interested to better understand the influence and perception of faces, to propose a visual saliency model with face features. Throughout the thesis, we concentrate on the question, "How people explore dynamic visual scenes, how the different visual features are modeled to mimic the eye movements of people, in particular, what is the influence of faces?" To answer these questions we analyze the influence of faces on gaze during free-viewing of videos, as well as the effects of the number, location and size of faces. Based on the findings of this work, we propose model with face as an important information feature extracted in parallel alongside other classical visual features (static and dynamic features). Finally, we propose a multi-GPU implementation of the visual saliency model, demonstrating an enormous speedup of more than 132× compared to a multithreaded CPU.

Résumé

Les études menées dans cette thèse portent sur le rôle des visages dans l'attention visuelle. Nous avons cherché à mieux comprendre l'influence des visages dans les vidéos sur les mouvements oculaires, afin de proposer un modèle de saillance visuelle pour la prédiction de la direction du regard. Pour cela, nous avons analysé l'effet des visages sur les fixations oculaires d'observateurs regardant librement (sans consigne ni tâche particulière) des vidéos. Nous avons étudié l'impact du nombre de visages, de leur emplacement et de leur taille. Il est apparu clairement que les visages dans une scène dynamique (à l'instar de ce qui se passe sur les images fixes) modifie fortement les mouvements oculaires. En nous appuyant sur ces résultats, nous avons proposé un modèle de saillance visuelle, qui combine des caractéristiques classiques de bas-niveau (orientations et fréquences spatiales, amplitude du mouvement des objets) avec cette caractéristique importante de plus haut-niveau que constitue les visages. Enfin, afin de permettre des traitements plus proches du temps réel, nous avons développé une implémentation parallèle de ce modèle de saillance visuelle sur une plateforme multi-GPU. Le gain en vitesse est d'environ 130 par rapport à une implémentation sur un processeur multithread.

Acknowledgments

I would like to express my deep gratitude to Prof. Dominique Houzet and Prof. Denis Pellerin, my thesis advisors, for their patient guidance, enthusiastic encouragement and constructive criticisms during the planning and development of my research work. I greatly appreciate their willingness to give their time so generously, and I look forward to collaborate with them in the future.

I would like to thank my thesis committee, Prof. Alain Tremeau, Dr. Simon Thorpe, Prof. Christopher Peters and Prof. Michel Paindavoine, for their insightful and extensive comments on my thesis manuscript and presentation. My grateful thanks are also extended to Dr. Nathalie Guyader, Dr. Sophie Marat, Dr. Vincent Fristot, Dr. Vincent Boulos, Dr. Sibt ul Hussain and Dr. Mian Muhammad Hamayun for their valuable advice and assistance throughout my research work.

I owe my thanks to Dr. Ziauddin Zia for teaching me the importance of learning. He helped me realize the potential in studying computer science, not only to strengthen my understanding of the field, but also the world around me. He continues to serve an important role-model for me.

I would like to express my thanks to all my Pakistani friends, as well as to all my colleagues with whom I had the opportunity to work with for three years. Moreover, I would like to mention that the research reported in this dissertation was funded by the Higher Education Commission (HEC), Pakistan.

Finally, I wish to thank my Father, my Mother, Hafsa, Nayab and Muneeb for their support and encouragement every now and then.

Thank you, Thank you, Thank you!!!

Contents

List of Figures

List of Tables

Publications

Some ideas and figures presented in the thesis have appeared previously in the following publications:

Refereed Journals

[Mar+13] S. Marat, A. Rahman, D. Pellerin, N. Guyader, and D. Houzet. "Improving visual saliency by adding 'face feature map' and 'center bias'". In: *Cogn. Comput.* 5.1 (2013), pp. 63–75.

[Rah+11a] A. Rahman, D. Houzet, D. Pellerin, S. Marat, and N. Guyader. "Parallel implementation of a spatio-temporal visual saliency model". In: *Journal of Real-Time Image Processing* 6.1 (2011), pp. 3–14.

Conferences

[HHR10] D. Houzet, S. Huet, and A. Rahman. "SysCellC: a data-flow programming model on multi-GPU". In: *Procedia Computer Science*. Vol. 1. 1. May 2010, pp. 1029–1038.

[RPH12] A. Rahman, D. Pellerin, and D. Houzet. "Face perception: Influence of location and number in videos". In: *Image Analysis for Multimedia Interactive Services (WIAMIS), 2012 13th International Workshop on*. Dublin, Ireland, 2012, pp. 1–4 (see p. 37).

[Rah+10] A. Rahman, D. Houzet, D. Pellerin, and L. Agud. "GPU implementation of motion estimation for visual saliency". In: *Proceedings of the Conference on Design and Architectures for Signal and Image Processing (DASIP 2010)*. Edinburgh, United Kingdom, Oct. 2010, pp. 222–227.

[Rah+11b] A. Rahman, G. Song, D. Pellerin, and D. Houzet. "Spatio-temporal fusion of visual attention model". In: *Proc. European Signal Processing Conference (EUSIPCO)*. Barcelona, Spain, 2011, pp. 2029–2033.

Book Chapters

[RHP11] A. Rahman, D. Houzet, and D. Pellerin. "Visual Saliency Model on Multi-GPU". In: *GPU Computing Gems Emerald Edition*. Elsevier, 2011, pp. 451–472.

A problem well stated is a problem half-solved.
Charles Kettering

Chapter 1

Introduction

Human beings have five traditional senses. Visual, auditory, olfactory, gustatory and tactile information each coming from a dedicated sensory system or organ. Only a fraction of this incoming information is selected through a well-developed and sophisticated selection process known as *attention*. Two factors are thought to drive the selection: (1) the inherent ability of the stimuli to capture attention, and (2) the information's behavioral importance to goals or tasks at hand. Primarily, the function of selection is to dedicate the limited resources only for significant information.

Human visual system is believed to be responsible for 90% of the information useful for perception, and hence, considerably important to function in daily-life. However, our visual system has limited capacity to process all the information falling on retina. To function and to complete tasks, it is important to extract a relevant portion of this information. This specific type of selection directing of computational resources to a small region, analogous to a spotlight, is called *selective visual attention*. The spotlight of focus acts more than the selection of a region of interest, but it is either captured by the low-level features of a region, or is drawn towards an object based on the task demands. It involves both bottom-up and top-down mechanisms, and their interactions making attentional capture work by planning and executing eye movements.

1.1 Challenges

The notion of selective attention is becoming important in the field of computer vision, and is an active area of research. The main purpose of selection of a region of interest is the omission of unwanted and redundant information, to increase the visual processing quality, while consuming less computing resources. It is a collaboration of earlier, parallel, exogenous processes with coarser, serial, endogenous mechanisms. The mechanisms are extremely efficient, but they are quite complex to imitate for artificial systems. Therefore, extensive studies to increase our understanding of visual attention is required.

Over the years, several models have been proposed for the pre-attentive or exogenous processes. They mostly operate on low-level features, for example: intensity, color, orientations, motion, etc. The values of these features determine the interest of any region in

a visual scene. Apart from these features, *faces* can deliver the most relevant information due to their social and evolutionary importance. Information regarding age, sex, race, emotions, attractiveness, and gaze direction from the facial information can guide attention. Some studies have found the faces to be processed during the early processing stages similar to other features. Moreover, the evidence of specialized neuronal circuits strongly suggest their capability of attentional capture, irrespective of the task demands. We consider that all these factors contribute to an emphasis on faces; that is, allocation of more attentional resources for faces by the visual system.

The first challenge, in this thesis, is to investigate the interest of faces using experiments and evaluations. There are several questions regarding whether allocation of attention on face is a function of its location, number, or size. They are arguably special, but what is the impact of these attributes? How the attention is allocated, when a face is encountered? Also, what can they convey? Is the interest task-dependent or not? How meaningful face signal is required for allocation of attention? What type of face captures attention?

The second challenge is to incorporate the findings from the experiments and evaluations into a computational model, or *visual saliency model*. In this thesis, the model studied is bottom-up that computes visual saliency of a scene. It outputs a map with all the salient regions highlighted, which can be used to extend and reduce the complexity of artificial vision algorithms. The idea here is to preprocess the visual information, and then to apply the computation to only the important or salient chunk of information.

The third challenge is with the inclusion of new features and improvements, which will increase the complexity of the existing models. Hence, it is important to compute them efficiently, and to be used by researchers. The objective here is to quickly compute and evaluate the eye movement predictions. A popular method is to propose a parallelized version of the model, which takes into account the speed factor, as well as the affordability and power consumption. Ultimately, the goal is to build a visual saliency model that is fast, accurate, robust, manageable and accessible.

The primary objective of the thesis is to study the interest of faces, and to incorporate a dedicated face channel into a visual saliency model. Ultimately, an efficient implementation of the model through parallelization will be useful for real-life vision systems.

1.2 Objectives

Based on the mentioned challenges, we clearly find the problem to be broad, and the solution involving different fields of science from cognition, computational modeling, and computer vision.

The first step is to perform human psychophysics experiments involving quantitative behavioral studies. The studies identify different features processed by underlying attentional mechanisms in the human brain. It also addresses the relevance of the common features identified to perception. In the end, the findings from such studies can guide and build a model for visual attention.

The second step involves computational modeling, to develop a description of the observed behavior of humans visual attention. The resulting model is either evaluated against experimental recordings of real subjects' eye movements, or compared qualitatively to other models. This generalization of the mechanisms deploying attention is important to study the interest of different features in the visual scene. The output of the models is often a saliency map with salient regions, which can have extensive implications in computer and machine vision. For example, in cognitive robots, an artificial model of human attention

can be used to make real-time cognitive decisions based on selective information from the surrounding environment.

The last step implements a vision system that works on low-resolution by the use of selective attention as a preprocessing stage. It is important that the selection stage is real-time or relatively faster, for example for autonomous robots and vehicles, object recognition, target tracking, security monitoring and industrial inspection. All these applications take advantage of selection in order to get efficient and effective resource allocation and utilization; necessarily only using the relevant information.

1.3 Main Contributions

The thesis relies on previous works on modeling of visual saliency, and make use of an already compiled video database and the corresponding experimental eye movements using an eye tracker. The proposed model is extended to increase its predictability. Following are main contributions made in this thesis.

- We investigate the influence of face on gaze in videos, with respect to location and number of faces. We used data from an eye movement experiment, and hand-labeled all the faces in the videos watched. We compared the eye positions and location of faces using different criteria. For the analysis, we considered frames with either one or two faces. In both cases, the scores decrease with increasing eccentricity. However, in the case of two faces, the individual score of the face at lower eccentricity is higher compared to its counterpart. In addition, we analyze fixation durations for one and two faces. We find that faces in videos lead to long fixations, considerably longer in the case of one face compared to the case of two faces. Furthermore, in the case of one face the fixations at low eccentricities last longer compared to those in the case of two faces.

- Faces play an important role in guiding visual attention, and thus, the inclusion of face detection into a classical visual attention model can improve eye movement predictions. In this thesis, we proposed a visual saliency model to predict eye movements during free viewing of videos. The model is inspired by the biology of the visual system and breaks down each frame of a video database into three saliency maps, each earmarked for a particular visual feature. (a) A 'static' saliency map emphasizes regions that differ from their context in terms of luminance, orientation and spatial frequency. (b) A 'dynamic' saliency map emphasizes moving regions with values proportional to motion amplitude. (c) A 'face' saliency map emphasizes areas where a face is detected with a value proportional to the confidence of the detection. Here, the first two channels are borrowed from Marat's visual saliency model [Mar10; Mar+09]. We compared the eye movements with the models' saliency maps to quantify their efficiency. We also examined the influence of center bias on the saliency maps and incorporated it into the model in a suitable way. Finally, we proposed an efficient fusion method of all these saliency maps. The fused master saliency map developed in this research is a good predictor of participants' eye positions.

- The proposed visual saliency model is complex and compute-intensive, and it requires a faster implementation to analyze the results as early as possible to efficiently move forward to analyze the predictability performance of the model. We propose a very efficient implementation of this model with multi-GPU. We present the algorithms of the model as well as several parallel optimizations on GPU with higher precision and

execution time results. The real-time execution of this multi-path model on multi-GPU makes it a powerful tool to facilitate many vision related applications.

1.4 Thesis Organization

Outline of this thesis is as follows. Chapter 1 introduces the scope of the thesis and highlights the different challenges, and the contributions made to tackle them. Chapter 2 brings the background material and forms the basis for next chapters. Background literature is reviewed from two perspectives: a) visual attention and face perception, b) and visual saliency modeling and its application. Chapter 3 explores the preference of faces in human visual system by analyzing eye fixations and saccades. This chapter forms a basis for Chapter 4 for saliency modeling with a dedicated face pathway. Chapter 5 summarizes our implemented parallel algorithm with different optimizations made. We report speedups, profiling results, and evaluate their validity. In Chapter 6, we conclude and give several perspectives.

I have found that the greatest help in meeting any problem is to know where you yourself stand. That is, to have in words what you believe and are acting from.

William Faulkner

Chapter 2

Background

2.1 Human visual system

HUMAN VISUAL SYSTEM (HVS) IS IMPORTANT as it account for 90% of the information coming from the five senses. Therefore, it is a crucial mean to interact with the environment, such as perception of objects, interpersonal communication and social interaction. HVS (Figure 2.1) can be described as consisting of two functional parts: a sensory system and a perceptual system, simply the eyes and the brain. The former is analogous to a biological camera performing the initial preprocessing and compression of the incoming visual information, while the later system performs complex operations on it.

The light entering the eyes first hits the cornea. It refracts, and passes through the aqueous, iris, lens and vitreous mucus to hit the wall of the retina, where the image is perceived. The retina contains 100 million rods and 6.5 million cones. Rods provide low-order illumination, whereas cones provide higher-order illumination responsible for color vision. The cones are mostly concentrated in the center, called the fovea. The concentration in highest within 1° of the fovea, and it drops considerably outside. To effectively process the incoming visual information using limited resources, HVS employs specialized mechanisms to effectively explore the entire visual scene within a small number of eye movements.

2.2 Human visual attention

Visual attention is to allocate visual processing resources to only the selected aspects of the visual scene, to gain as much information as possible. The main goal is to function better by faster and more robust visual cognitive processing of the information. For example, it is also important to detect a change or an event, and resultantly allocate attentional resources towards that change.

The different regions involved in driving visual attention are: visual cortex, inferotemporal cortex (IT), posterier parietal cortex (PPC), prefrontal cortex (PFC) and superior colliculus (SC) [Pal99; BI11]. The visual information enters through the visual cortex (V1, V2, V3, V4 and V5/MT), bifurcates into two separate pathways—ventral and dorsal streams. The dorsal stream's main function is fixating the region of interest, and occurs in the PPC. This

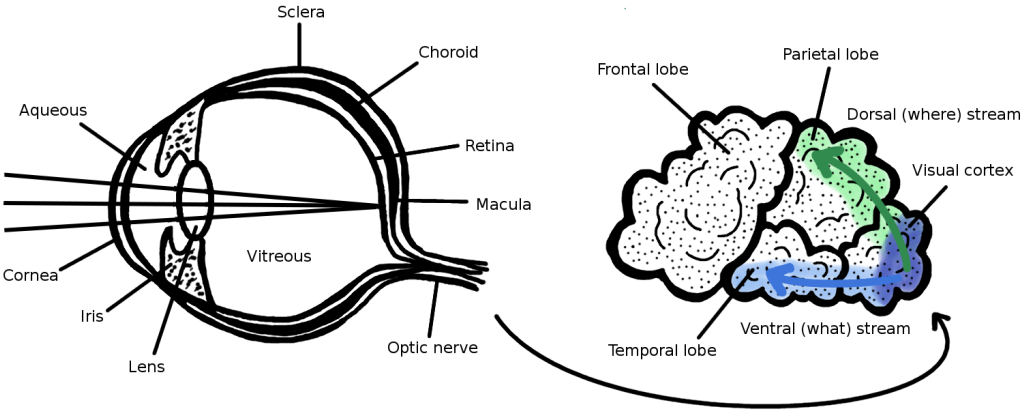


Figure 2.1: Human visual system is composed of two functional parts: a sensory system (eye) and a perceptual system (visual areas).

suggests the independence of dorsal stream from goal-driven control; that is, the attention is directed using low-level features. On the other hand, the ventral stream is connected to the IT involving higher mechanisms. PFC is bidirectionally connected to PPC and IT, and it acts as a modulator of the two streams. Aside from the modulation function, PFC also controls the eye movements through the SC.

The main functions of visual attention include:

- ❖ Selection of the region of attention. Brain is not capable of processing the entire incoming information, hence selective attention is used to select and prioritize information. These mechanisms provide information manageable, computationally efficient, localized, and relevant to the task demands.
- ❖ Extraction of different features from object stimuli, their organization by higher mechanisms for object recognition, and ultimately resolution of their value.
- ❖ Manipulation of incoming information. Human vision is not merely limited to selection, but involve more complex and highly parallel mechanisms working with more coarser information for detailed analysis of the visual scene.
- ❖ Initiation of attentional shift by planning eye movements combined with knowledge, expectations and task demands. The movements comprise of saccades and following fixations, when aggregated result in a complete scan path.

Visual attention is thought to be driven by two types of processes. Exogenous or bottom-up visual attention—intrinsic, automatic and subconsciously performed eye movements—are mostly influenced by the low-level features of the stimulus independent from high-level mechanisms [The90; The94; YJ96; PN04]. The pre-attentive selection of regions of interest, or the salient regions is important due to the limited information carrying capacity of the optic nerve [PLN02], and lasts for only $25 - 50ms$. On the other hand, endogenous or top-down attention is based on the task demands or expectations, irrespective of the features of the objects in the visual scene [FRJ92; FRW94; BE94]. The mechanisms are longer around $200ms$.

Both bottom-up and top-down components are important, as experiments with tasks can cause voluntary control immediately after the visual scene onset. This is because of the interactions between stimulus-driven properties and goal-driven mechanism, both

complementing one another in selection and deployment of attention by making eye movements.

Many theories and hypothesis regarding the working of visual attention in primates have been proposed. Nevertheless, there are still plenty of open questions to answer, involving fields ranging from biology, psychology, neuroscience and computer science.

2.3 Eye movements

Eye movements are important in natural and complex scenes [Yar67]. The pattern of movements exhibited is cognitive and task dependent, for example, usually the most salient region is fixated during free-viewing of a visual scene. Moreover, the number of fixations on a region determines its informativeness [SFH86; HS95; DS96; BS97; MMN99].

In case of stationary objects of interest, eye movements follow a two-pass location perception model [AKH93]. First a movement based on its coarser location information, and then followed by small refined movement to fixate the object. The resulting foveation is important to collect more detailed information. There are influences of both stimulus-driven and goal-driven attention on eye movements. In early vision, the former is thought to devise a model of the visual scene, which acts as a precursor to plan eye movements to complete the task.

On the scene onset, there is a delay of about $200ms$, the initial saccades planning time [FW93; Hof98]. This delay is immediately followed by jumpy eye movements lasting $20 - 200ms$, called saccades, towards the locality of the salient region. The initiation and planning of saccades is important for understanding the importance and working of visual attention [HS95; Hof98]. The initial saccade with higher velocity is followed by decelerated ones. This deceleration is attributed to the more information to collect, and consequently use to direct the eye movements towards the salient regions. Only a small portion of the visual field is attended through the fovea (around 2 to 3°), which lasts for about $250-500ms$ [Viv90]. To explore the scene, a series of foveations are required, called fixations. These aggregated with the saccades result in a scanpath [NS71] (Figure 2.2).

The distance traveled by a saccade is its amplitude, varying around 0° to 40° with a peak velocities from 30 to 700°/s. The saccade duration depends on the distance covered, as well as, the viewing conditions that include the screen size, visual quality, etc. As the amplitude of the saccades increases, it is considered that most of the visual scene is ignored.

Visual attention can be either overt or covert. Overt when the sensory organs are directed towards the focused stimuli, whereas covert when possible candidates to attend are mentally focused. The later is linked to the neuronal circuitry responsible for planning the shift of gaze. It is a mechanism making quick scans of the entire scene for potential regions of interest, and then to facilitate in setup of slower saccades towards the selected region. Different studies have found covert attention precede overt attention, which is the region where the attention subsequently lands. Therefore, visual attention is capable of anticipating the next eye movements using the limited information about the potential targets.

The eye movements recording (eye fixations and saccades) during psychophysics experiments are used to determine relationship between the responses of the participants and stimuli presented on the screen. The relationship can help understand the behavioral choices made depending on information acquired, location of attention, emotional state, or task-related.

Figure 2.2: A scanpath during viewing of a dynamic scene (illustrated as overlapped first,middle and last frames). The circles are the fixations connected by preceding saccades, while the diameter of the circles corresponds to fixation duration. We observe that the entire scanpath remain on the object of interest in the scene, the face.

2.4 Saliency-based model of visual attention

The very first model was proposed by Triesman and Gelade [TG80], called feature integration theory (FIT). The theory was developed using findings from experiments involving visual search. The first finding, addresses pop-out factor of a target when it is surrounded by distractors; that is, the participants show constant reaction times. The second finding, observes that when a target is discriminated based on combination of features, it shows linear increase in reaction times with increasing number of distractors. FIT models these findings into a two-step strategy called unified perception. The strategy starts with the division of information into distinct subsystems, which is analyzed for the properties. In the end, the information is combined together.

Different psychophysics studies and experiments conclude that attention follows two-step strategy. The first step is parallel operating on the entire visual field, while second step is sequential selecting only the relevant regions. Using this theory numerous models have been developed, in other words, such models find, code and unify feature characteristics using different methods. On example of such models is Guided search model proposed by Wolfe [WG97]. The model uses retinotopic-like feature maps to get a saliency map, which then drives visual attention in a serial fashion across the different salient regions.

The biologically plausible computational model for visual attention is based on neuro-physiological processes in human vision system. Such models are stimulated by similar visual features in the visual scene as in primates, and in consequence attracting focus. The very first biologically plausible model is proposed by Koch and Ullman [KU85]. The model takes into account the entire visual field. It is stimulus-driven using basic features like color and orientations to get a saliency map. The map represents the locations highly influenced by saliency of their surrounding regions. Each highly salient region is visited for spotlight or attention in sequence using inhibition of return. Necessarily, the model works in three stages: separate and code different features in parallel, combine the feature maps to get saliency map, which is used to direct attention in serial to different locations.

The first implementation of the model [Cha90] uses sets of feature maps to compute activation maps, a result of thresholds rather than lateral inhibitions proposed by Koch and

Ullman. Another model named selective tuning model [CT92] skips the computation of saliency map, and uses a hierarchically organized winner-takes-all (WTA) approach. The location with activity is beamed from the top level, moving through the inner layers of the pyramid, and at the lowest level corresponds to the focus of attention. Milanese [MGP95] implemented all three stages of selective attention model, while Itti [IKN98] proposed a more computationally efficient version using a multi-scale hierarchically structured approach for computation of feature maps. The later model is quite popular in the community. It extracts information from color, orientation and intensity cues, and uses a WTA approach to select the salient region. All these models have been compared with psychophysics studies, and it has been demonstrated that the three-stage model for selective attention works.

Traditionally, it is thought that bottom-up mechanisms only processes very basic visual features, while top-down mechanisms are behind the processing of complex features. The boundary between the two mechanisms is somewhat blur because recent studies [HH05] show that early pre-attentive mechanisms can process complex stimuli such as socially meaningful faces. Based on this fact, different bottom-up visual saliency models are augmented with face features for still images [CFK09; SCH09; Ton+10; WWZ10; SS12]. The results show an improved predictability of eye movements.

The models mentioned are efficient to compute because they involve low-level attributes with no learning phase. But, in case of high-level tasks, for example object detection and recognition, the models cannot differentiate between salient information in foreground or background. Hence, top-down modulated saliency model outperforms pure low-level stimuli-driven model to predict eye positions. Initially, task-related influences were added to the spatial-based attention model as a simple top-down component [CF89; CW90], while others used the knowledge from the scene to set the parameters for feature maps [OAVE93; PS00]. Recent models combine both bottom-up and top-down information to modulate, or assign weights to the salient regions [Tor+06; Kan+09]. In these models, target features combined with low-level features are used to guide eye movements. Overall the predictability of human vision system can be improved by taking advantage of multiple sources of information.

As the methods in computer vision are becoming more and more complex every day, a preprocessing stage using visual saliency is becoming popular—to limit the amount of information to process. The approach is potentially very interesting in a lot of applications because the use of relevant information based on saliency get improve the performance of the goal. Example of such applications include: image retargeting [Liu+10; Fan+12], compression [MZ08; HB11], recognition [Rut+04; GHV09; RAK09], automatic target detection [SM10], and many more. To make this possible, it is important to understand the mechanisms behind visual saliency, and devise a biologically plausible model. This might involve people doing psychophysics and neurophysiology experiments, as well as computational modeling. Although there are plenty of studies about the underlying mechanisms of visual attention. However, a complete neuromorphic bottom-up or top-down model is not yet unraveled.

2.5 Influence of faces on gaze

Faces play an important role in guiding visual attention, and they immediately attract the eyes when people are looking at static images [Cer+07]. Their explicit representation, speedier processing and automatic attentional shifts without endogenous control [Dri+99; LB99] strongly suggests that faces are preferred by primates. The preference might be linked to the social and biological importance of faces, or the information conveyed by them, such

as eye gaze, visual speech, and facial emotions [And98; BBK08]. Over the years, many studies provide evidence for the existence of several perceptual and attentional face processing mechanisms.

Information about the presence of a face can be extracted shortly after scene onset [Liu+09]. Different neuroimaging studies claim that face detection is coded in a specific cortical area of the brain; the fusiform face area (FFA) [Mec+04; KY06; Sum+06]. Similarly, electrophysiological studies show that face processing is remarkably fast, and human faces evoke a negative potential around 172*ms* (N170) [Ben+96]. Others have found the early neuronal face-selective responses to occur around 100*ms* [LHK02; CKT10], and in one case as early as 70*ms* [OP92; Geo+97; BBS01]. Thus, face information can control the initial eye movements.

Face being perceptually important [JM01; Tau09] also exhibits some attentional preference [FW+08]. Many studies have found it to be behind the early-onset responses to face stimuli [RRL01; Vui00]. In natural scenes, it can pop-out. It is fixated for longer intervals compared to the rest of the body [BBK08; BBK09]. Any changes in it can be detected rapidly [RRL01], and it is more effective when with emotions [MB99; Fox+00; OLE01; Com03; FG05]. This advantage of face is thought to be caused by some holistic face processing mechanism [HH05; HH06] in HVS that can process faces as a whole in parallel. The automatically processed face information is used to complete the task at hand. As a consequence, it leads to a slight delay in the deployment of complete endogenous or voluntary control for attention [MF88; MR89; SM89; CL91].

A face is preferred when presented alongside other objects because they are processed differently. Interestingly, they are difficult to ignore when presented as distractors. On the contrary, other object-stimuli can easily be avoided when the target is a face. In a relevant categorization experiment [Bin+07], face and object stimuli are presented in opposite cue locations—to the left and right of fixation. The response times towards the target cue locations are faster when it is a face, necessarily due to the attentional bias for faces. However, when the participants are asked to ignore faces completely, the response times towards object cue locations became faster. Furthermore, an upright face is processed effectively from distractors compared to line-drawn, inverted, scrambled, or animal faces [BHF97]. A study [Ros+00] about face inversion effect [Yin69] shows that there is an impact on recognition task performance when face stimuli is inverted. This is not the case for other object stimuli.

In conclusion, there is evidence that faces are special, and they are preferred when attentional resources are limited. There are many studies done that suggest the existence of both exogenous and endogenous processes in HVS to process faces. However, it is not yet completely clear, what exactly are the underlying perceptual and attentional mechanisms for faces, and how they process faces?

2.6 Interest of high-performance computing

Parallelization of an algorithm is in fact not a straightforward solution. It demands expertise to parallelize algorithms, as well as requires considerable effort depending on the target architecture. To cope with increasing computational demands of algorithms, designers have come up with many variations of hardware with different capabilities. The main goal is to achieve fast computation times, while keeping the problems of memory and power at bay. But, the variation across different platforms makes parallelization a challenging task. Thus, it is important to choose a platform based on the requirements of the algorithm, and the time constraints related develop and optimize accordingly.

In most cases of image processing and computer vision algorithms, there is an abundance of independent computations on every pixel. This concurrency or data parallelism can be exploited, but some algorithms might be strictly serial. The case includes algorithms using data-based on some priority, or every single iteration of computation is dependent on the previous iteration – iterative algorithms. In other words, it is important to have enough independent computing steps that can be mapped on to all the available processing units, necessarily use all the computational power available.

Two points are important when parallelizing an algorithm: first, it is crucial to minimize and optimize memory access operations because memory bandwidth could not keep up with the increasing trend of hardware vendors adding processing units on a single die. In a nutshell, the objective is to achieve a balance between compute and memory operations. Second, aside from just parallelizing the algorithms, a significant improvement sometimes requires reformulation of the existing problem, proposing an innovative algorithm that scales better on parallel machines.

The motivation here is to model HVS, and use it in real-life applications. The implementation of the model can be considered real-time only when it can process input efficiently to make decisions, or to guide a control in real-life scenarios. Inherently, the algorithms are compute-intensive, and exhibit parallelism. Hence, similar to other computer vision algorithms, they are natural candidates of high-performance computing.

2.7 Relationship to Past Work

The thesis presents a visual saliency model based on one proposed by Marat in her thesis [Mar+09; Mar10]. The model decomposes visual information into two channels: static and dynamic. Both these channels are treated independently with several common modules. This processing to extract salient information is somewhat bio-inspired. In the final stage, the saliency maps from the two channels are fused together to get a master saliency map. The thesis also proposed a three pathway visual saliency model after taking into account that face can improve the predictions of the visual model.

Many recent works have successfully incorporated a face channel into different models for still images [CFK09; SCH09; Ton+10; WWZ10; SS12]. All these studies use different methods to detect faces depending on requirements of the applications, in some cases a skin-color model might work well, while in other situations a complex face detector could be necessary.

Since many of the previous works use still images, in this thesis, we argue that faces are preferred in dynamic scenes. We expect that there is an improvement in predictions with the inclusion of a face channel into our visual saliency model. Also, it is important to have different channels, to make the model compute saliency of varying stimuli. We evaluate the model against a couple of video databases. The main objective is to test the consistency of the visual saliency predictor. In summary, our original contributions relate mainly to the face channel and parallelization of the entire model. This will help to efficiently understand the complex process of human visual attention.

A man's face as a rule says more, and more interesting things, than his mouth, for it is a compendium of everything his mouth will ever say, in that it is the monogram of all this man's thoughts and aspirations.

Arthur Schopenhauer

Chapter 3

Face Perception in Videos

Gaze is highly influenced by faces in visual scenes compared to other object stimuli. Several studies have reported the preference of faces in static images, and their influence on gaze. Contrarily, the influence of faces has rarely been reported for dynamic stimuli. In videos, object stimulus patterns degrade due to the loss of information as it moves away from the foveal region. This degradation of information seems likely to influence the preference of faces in videos. In addition to effects of location in a scene, the number of faces also limits the preference of faces, as they compete for the limited attentional resources. Both these limiting factors can be alleviated by the size of faces, which causes stimulus magnification to maintain foveal performance of faces, and to diminish the effects of competition.

The purpose of the current chapter is to study the influence of faces on gaze during free-viewing of videos, and analyze the effects of the number, location and size of faces. We hypothesized that faces have a preference in videos. It is preferred even at large eccentricities unless there are no other competing faces. The study reported examines the different influencing factors: number of faces, eccentricity of faces and area of faces, and tests the hypothesis by analyzing different combinations of these influencing factors using several evaluation criteria and other eye movement attributes like fixation dispersion among participants, fixation duration and fixation proportion. The findings from this work could support the possibility of adding a separate face pathway to a visual saliency model—to accurately predict eye movements.

3.1 Related work

Information about presence of faces can be extracted shortly after scene onset [Liu+09]. Different studies claim that face detection is coded in a specific cortical area of the brain; the fusiform face area (FFA) [KY06]. Related electrophysiological studies show that face processing is remarkably fast, and human faces can evoke a negative potential around $172ms$ ($N170$[1]) [Ben+96]. Some studies found these early neuronal face-selective responses

[1] The N170 is a component of the event-related potential reflects the neural processing of faces. It is an increase in negative potential over electrodes located at the fusiform and inferior-temporal gyri, when face images are presented.

occurring around 100*ms* [CKT10], while in one case as early as 70*ms* [BBS01]. This explicit representation of faces allows to control the eye movements, and hence, suggests their importance for primates during social interaction [And98].

Over the years, a number of studies have been conducted regarding the influence of faces on gaze, mostly using static stimuli [Lev+01; MLH02; Has+02; RMFT03; Rou+05; Jeb+09; FRW10; JR04; JR06; BSB09; HS10]. There is enough evidence that faces can be processed at the earliest after stimulus presentation [RRL01; Vui00], and they are preferentially processed by the human visual system compared to other object categories [Ros+00]. The preference is thought to be influenced by several different factors like face eccentricity from fixation, face area, and closeness to other faces.

Degradation of information in periphery has been studied thoroughly in the literature. It is most likely linked to the size of underlying processing mechanisms [Cur+90]; that is, early visual areas in primates are retinotopically organized with center-periphery organization extending well beyond the retina into higher visual areas. Evidence shows that the accuracy and quality of visual performance in the periphery is identical to one in the foveal vision [STB89; BSA91], but the drop is due to progressive under-sampling of information presented away from fixation. Consequently, the degradation of details limits the capacity of human visual system to extract information, which is important to attend objects in a natural scene. Over the years, the influence of peripheral vision on faces have been thoroughly studied, with controlled presentation of visual stimuli at predefined locations on rings of different eccentricities [Par+03; RRK06; Jeb+09; Her+10; Rig+11]. Most found a drop in performance of object stimuli, in the case of faces in periphery, a steep drop in face-selective responses. The main question in this study is whether eccentricity-dependent sensitivity loss of faces occurs when viewing dynamic stimuli.

The direct effect of eccentricity on stimuli can be compensated using some linear eccentricity-dependent magnification, or size scaling [VR79; Dow+81; LKA85; JG10]. A single magnification factor based on eccentricity will fail to compensate for eccentricity-dependent loss depending on the task demands, and increased interference caused by visual crowding with eccentricity [Mel+00; CLL07; PPM04].

Event-related response to face stimulus, the N170, reduce considerably when more stimuli are presented in the visual field [MGG93; RT95]. This suppression of neural representation for stimuli is referred to as competition [KU01; JR04; JR06]. A recent study [JR04] showed that the response to foveal faces is reduced when another face is presented parafoveally. The suppression remained even when a scrambled face was presented as a competing stimuli. This suggests that there is certainly some sensory competition rather than simply an effect of reduced spatial attention.

Faces are equally distinguishable in the periphery, as they are in the periphery, unless there are no other competing faces [FRW10]. In the later condition of competing face stimuli, the foveal face gets more competing advantage against to the underrepresented peripheral faces [JR06]. When face stimuli are presented in the periphery, the suppression effects on faces disappear, and the competition is modulated by a foveal bias [Rou+05].

In the chapter, we first describe the eye movement experiment and video database used (Section 3.2). To study the influence of different factors (Section 3.3), such as eccentricity, area and number of faces, we evaluated eye fixations data for scenes with faces using several evaluation measures. First, we analyze dispersion among participants when face is presented (Section 3.4.1), and distance of fixations made from the face Section 3.4.2. Second, we present the findings from comparison criteria for one face and two faces (Section 3.5). Last, we analyze the influence of the different factors on fixation durations. We conclude the study

using video database with a discussion in Section 3.7. Finally, using the findings of the work, we propose a modulated face pathway described in Section 3.8.

3.2 Eye movement experiment

We used the eye position data from a previous experiment described in [Mar+09]. The experiment aimed to record eye movements of participants when looking freely at videos with various contents. We used this data to understand the features that best explain eye movements and fixated locations. Here, we recall some of the main aspects of this experiment.

3.2.1 Video dataset

Fifty-three videos (25fps, 720×576 pixels per frame) were selected from different video sources, for example: indoor scenes, outdoor, scenes of day and night (Figure 3.1). The videos are converted to grayscale before presenting them to the participants.

Figure 3.1: Some examples of images from different video sources, for example: indoor scenes, outdoor, scenes of day and night.

The videos were cut into 305 *clip snippets* each of 1-3s. This was done in manner to get snippets with minimum change in plane. Finally, these *clip snippets* were strung together to obtain 20 *clips* of 30s, as shown in Figure 3.2. Each *clip* comprised of a *clip snippet* from every source. The duration of the *clip* was random to eliminate any transition anticipations made by the participants during viewing.

3.2.2 Participants

Fifteen young adults (3 women and 12 men, range 23-40 years) participated in the experiment. All participants had normal or corrected to normal vision. Each participant sitting with his/her head stabilized on a chin rest, in front of a monitor at 57cm viewing distance ($40° \times 30°$ field of view), was instructed to look at the videos without any particular task.

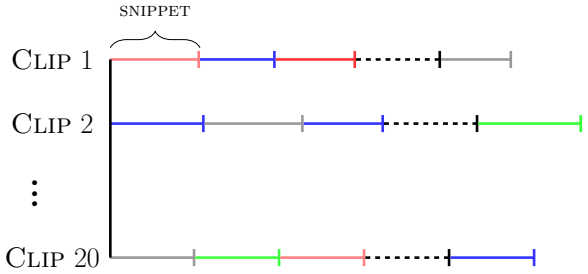


Figure 3.2: Videos were cut into 305 *clip snippets* of 1-3s, strung together to obtain 20 *clips* of around 30s.

3.2.3 Data acquisition

An eye tracker (SR Research EyeLink II) was used to record eye movements. It composed of three miniature cameras mounted on a helmet. Two in front of each eye to provide binocular tracking, while the third on a head-band for head tracking. The recordings from the former two cameras when compensated for head movements gives the gaze direction of participant.

3.3 Method

Faces are interesting in dynamic scenes, and they influence eye movements. In this study, we test a video database comprising faces to analyze their interest during free-viewing. We also evaluate the influence of different factors on the interest of faces, such as number of faces, face eccentricity, face area, and closeness between two faces. In this section, we first define these influencing factors. Second, we detail several evaluation measures to analyze the influencing factor. Third, we present the data used for the evaluation that includes the hand-labeled faces of the entire video dataset, and the eye fixations recorded during the eye movement experiment. Last, we summarize the methods used for statistical analysis of the data.

3.3.1 Influencing factors

The study was designed to provide an insight into the extent to which different visual factors of the faces affect their perceived interest. In addition to the number of faces in a scene, we annotate each face in the scene with its area and eccentricity, and closeness to other faces. These are measured as follows:

- **Number.** is a simple count of faces present in a scene. It determines the complexity of the scene. For clarity, we only consider cases of frames with one face and two faces.

- **Eccentricity** is the distance from participant's fovea to the edge of face ellipse in degrees. In Figure 3.3, $(d - r(\alpha))$ is the eccentricity of the face ellipse with origin (O_x, O_y) from the fixation position (C_x, C_y). To compute the eccentricity E of a face, we first compute the eccentric angle of the fixation position from the face ellipse.

$$\alpha = \arctan\left(\frac{b}{a}\right) = \arctan\left(\frac{|O_x - C_x|}{|O_y - C_y|}\right)$$

The distance of the origin of ellipse (O_x, O_y) from position at angle α on the ellipse is computed as:

$$r(\alpha) = \frac{r_a \cdot r_b}{\sqrt{(r_b \cdot \cos(\alpha))^2 + (r_a \cdot \sin(\alpha))^2}}$$

where r_a and r_b are the major and minor axis of the ellipse, corresponding to one-half of the width and height of the face respectively.

The distance from the fixation to the position on the face ellipse at radius $r(\alpha)$ is defined as the eccentricity E of the face from the fixation. It is computed by subtracting the radius $r(\alpha)$ from the euclidean distance d between fixation (C_x, C_y) and origin of face ellipse (O_x, O_y).

$$E = \begin{cases} d - r(\alpha) & \text{if } (r(\alpha) < d) \\ 0 & \text{otherwise} \end{cases}$$

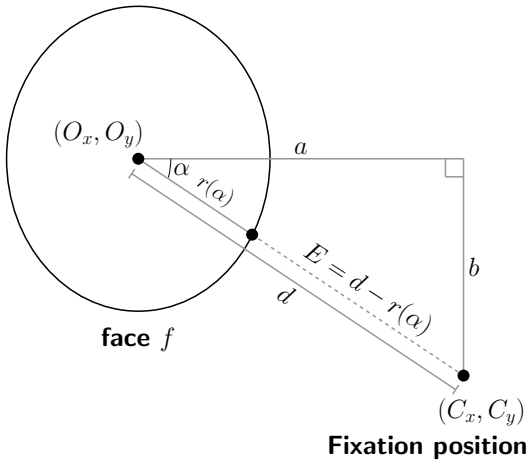


Figure 3.3: Eccentricity of face presented on screen from fixation. Consider a face ellipse f with major and minor axis, r_a and r_b, equal to face dimensions. $r(\alpha)$ is the radius to the position on the ellipse at angle α of a right angle triangle with legs of length a and b. The angle is measured from the major axis of the face ellipse f to the fixation position (C_x, C_y). Finally, the radius $r(\alpha)$ is subtracted from the euclidean distance d between the origin of the face ellipse (O_x, O_y) and fixation position to get eccentricity of the face.

- **Area** is the two-dimensional surface of face ellipse in squared degrees. It is calculated as πab, where a and b are one-half of the face ellipse's major and minor axes respectively.

- **Closeness:** between the faces f^1 and f^2 in the case of two faces is the euclidean distance between the two face regions. In Figure 3.4, $(d - (r(\alpha) + r(\beta))$ is the closeness between the face ellipses with origins O^{f^1} and O^{f^2}. To compute the closeness C of a face, we first compute the eccentric angle of the counterpart face's origin to the origin of the

face considered. Angle α for face f^1, and angle β for face f^2.

$$\alpha = \arctan\left(\frac{b}{a}\right) \cdot \left(\frac{180}{\pi}\right)$$
$$\beta = \arctan\left(\frac{a}{b}\right) \cdot \left(\frac{180}{\pi}\right)$$
$$\text{where, } \begin{cases} a = |O_x^{f^1} - O_x^{f^2}| \\ b = |O_y^{f^1} - O_y^{f^2}| \end{cases}$$

The distance of the origin of face ellipse from position at angle θ on the face ellipse is computed as:

$$r(\theta) = \sqrt{\frac{1}{\left(\frac{\sin\theta}{r_a}\right)^2 + \left(\frac{\cos\theta}{r_b}\right)^2}}$$

where r_a and r_b are the major and minor axis of the ellipse, corresponding to one-half of the width and height of the face respectively. The equation is used to compute the radii $r(\alpha)$ and $r(\beta)$ at angles α and β for the two face ellipses f^1 and f^2.

The distance between the respective positions on the two face ellipses at radii $r(\alpha)$ and $r(\beta)$ is defined as the closeness C between the two faces. The value is computed by subtracting the radii $r(\alpha)$ and $r(\beta)$ from the euclidean distance d between the origins of the face ellipses, O^{f^1} and O^{f^2}.

$$C = \begin{cases} d - (r(\alpha) + r(\beta)) & \text{if } (r(\alpha) + r(\beta) < d) \\ 0 & \text{otherwise} \end{cases}$$

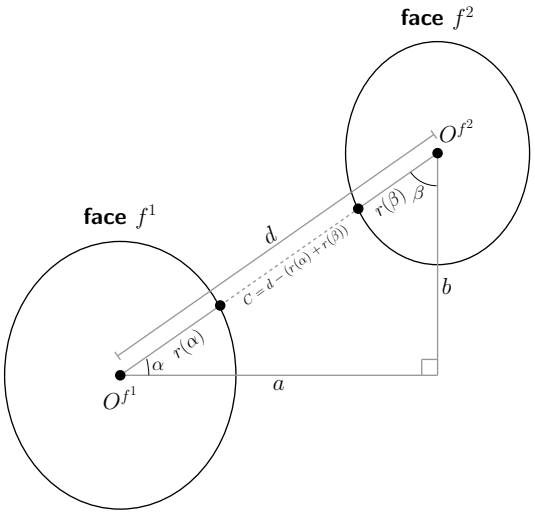


Figure 3.4: Closeness between two faces presented on screen. Consider two face ellipses f^1 and f^2 with major and minor axis equal to the respective dimensions of the faces. $r(\alpha)$ and $r(\beta)$ are the radii to positions on the ellipses at angles α and β of a right angle triangle with legs of length a and b. The angles are measured from the major axis of one face ellipse to the origin the counterpart face ellipse. Finally, the radii $r(\alpha)$ and $r(\beta)$ subtracted from the euclidean distance d between the origins O^{f^1} and O^{f^2} to get closeness between faces.

3.3.2 Evaluation measures

We used several evaluation measures to investigate the effects of faces on eye fixations during free-viewing of a visual scene. The first two measures mentioned below were used to confirm the interest of faces in dynamic stimuli, while the rest were used to investigate the performance of faces with respect to different influencing factors, such as number, eccentricity, area, and closeness of faces.

Inter-observer congruency

Inter-observer congruency (IOC), or *fixation dispersion* [TBG05] can be measured by using a *'one-against-all'* approach, also called *'leave one out'* approach [Tor+06]. It computes the degree of similarity between eye fixations of one participant to those of other participants. The final fixation dispersion among participants is obtained by averaging the degree of similarity over all participants for clusters C determined by the number of faces in the scene.

$$IOC = \|D_{x \in C}\|, \text{ with } D = \frac{1}{N^2} \sum_{i,j<i} d_{i,j}^2$$

where N is the number of participants and $d_{i,j}$ is the distance between the eye fixations of participants i and j. A lower value shows the eye fixations made to be closer for the participants.

Minimum fixation distance

Minimum fixation distance, or *Shortest Euclidean distance* [WP12] from faces in a scene to fixation, or fovea of the participant during a trial. The distance is computed from the fixation position to the face region of interest—the edge of face ellipse. Essentially, it is equal to the eccentricity E of the face closest to the fixation, as defined in Section 3.3.1.

$$d_{min} = arg\,min_E$$

In the case of scenes with one face, we measured the distance from the fixation to the face. Likewise, it is measured in the case of two faces; however, to the face with minimum eccentricity from fixation. The distances are computed from the fixation coordinates to the edge of the face ellipses.

Fixation proportion

We categorized the fixations on scenes with faces into two types: fixations landing inside a face, called *'on-face' (oF)* fixations, and fixations landing outside a face, called *'not-on-face' (nF)* fixations (Figure 3.5). This was done by comparing fixation coordinates to a face, represented by an elliptical mask equal to the face dimensions plus 1° of margin. Here, we did not consider fixations for scenes with no faces to fixate upon.

Similarity between maps

Different criteria are used to predict the likelihood of different regions attracting attention in a scene. It is often done by comparing such regions of interest to participants eye movements [IKN98; PLN02; TBG05; Pet+05; Tor+06; LM+06]. In the literature,

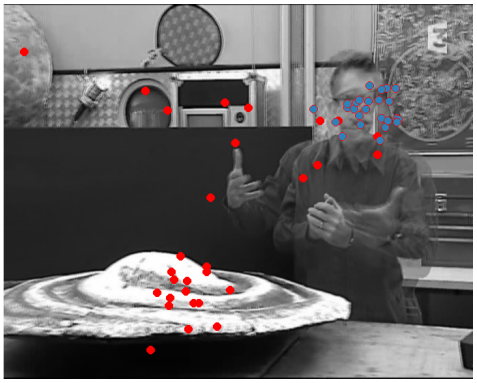


Figure 3.5: Superimposed frames of a video snippet with overlayed *'on-face' (oF)* and *'not-on-face' (nF)* fixations.

several criteria are proposed: ROC (receiver operating characteristic) [TBG05; LM+10], NSS (normalized saliency scanpath), Percentile, Kullback Leibler divergence [Pet+05] and rate of correct fixations [Tor+06].

In this study, we are interested in analyzing the influence of faces in a scene. We used different criteria to measure the correspondence between regions predicted to be fixated and regions fixated by participants, represented as face maps and eye fixations maps respectively.

- **Face maps:** We computed face map M^f (Figure 3.6b) for each frame by hand-labeling the position of the face using a bounding box, and then applying a 2D Gaussian to it. The dimensions of the bounding box determine the variance of the 2D Gaussian from origin in horizontal and vertical axis, whereas the amplitude of the function was kept constant for all faces. All values outside the elliptical face were set to zero.

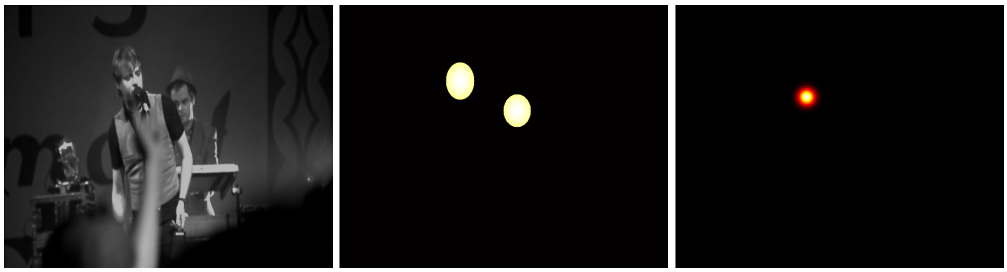
- **Eye fixation maps:** The eye fixation maps were defined for each fixation made by a participant. It is simply a Gaussian at the fixation position for one participant ($\sigma = 0.5°$), denoted as M^h. The maps were used to evaluate faces using some comparison criteria. A sample M^h map is illustrated in Figure 3.6c.

- **Comparison criteria:**

  To compute the comparison criteria, for instance for the first fixation, we compare M^h for each participant to all M^f maps for the entire duration of the fixation. The values are then averaged to get a score for the participant. Likewise, this process is repeated for all participants. Finally, all individual scores from all participants are averaged to get the score for the fixation. In this study, we used five comparison criteria to estimate the relevance between the eye fixation map M^h and face map M^f.

  - **AUC criterion:**

    The degree of similarity between the eye fixation map M^h and face map M^f has been computed through a receiver operating characteristic(ROC) analysis [Faw06]. It consists in estimating the true positive rate (TPR) and the false positive rate (FPR), by labeling each pixel of the eye fixation maps M^h as fixated or not fixated. The face map M^f is then treated as a binary classifier to separate the positive samples from the negatives. By thresholding over the face map M^f and plotting



(a) (b) (c)

Figure 3.6: From left to right (a) input frame with faces, (b) face map M^f for the input frame with 2D-Gaussian faces from the ground truth, and (c) and the frame's corresponding fixation with Gaussian for one participant ($\sigma = 0.5°$), referred to as M^h. To compute the comparison criteria, for instance for the first fixation, we compare M^h for each participant to all M^f maps for the entire duration of the fixation. The values are then averaged to get a score for the participant. Likewise, this process is repeated for all participants. Finally, all individual scores from all participants are averaged to get the score for the fixation. Note that in the case of face maps M^f, the dimensions of the face defines the standard deviations of the applied Gaussian where all values lying outside the resulting face ellipse are set to zero. Consequently, we get a face map with upper and lower extremities of the Gaussian curve removed.

true positive rate vs. false positive rate an ROC curve is achieved.

$$\text{TPR} \approx \frac{\text{positives correctly classified}}{\text{total positives}}$$
$$\text{FPR} \approx \frac{\text{negatives incorrectly classified}}{\text{total negatives}}$$

Area underneath the curve is referred to as AUC, or area under the receiver operating characteristic (ROC) curve. Usually it is calculated using the trapezoidal rule:

$$A = \frac{1}{2}\sum_{i=2}^{N}(x_i - x_{i-1}) \cdot (y_i + y_{i-1})$$

It is usually taken as a scalar value, such that a value $A = 0.5$ reflects random forecasts, whereas $A = 1.0$ implies perfect forecasts.

– **TC criterion:** Torralba et al. [Tor+06] proposed a method to evaluate the quality of a face map. It simply estimates the ratio of eye fixations predicted by the map over all face regions. A correct prediction is when a fixation projects onto a face region, which is 20% of the map surface. This criterion is easy to calculate, although it requires a threshold.

$$TC = 100 \times \frac{N_{within}}{N_{all}}\%$$
$$N_{within}: \quad \text{number of eye fixations within face regions}$$
$$N_{all}: \quad \text{total number of eye fixations}$$

– **NSS criterion:** Normalized Saliency Scanpath proposed by Peters et al. [Pet+05] is an averaging of the pixels that correspond to eye fixations. It acts as a z-score

computed by comparing a predicted face map to participants' eye fixations. The NSS value of face map M^f for an eye fixation at position (x_h, y_h) is:

$$NSS = \frac{1}{\sigma_{M^f}} \cdot (M^f_{(x_h,y_h)} - \mu_{M^f})$$

The NSS at any position is simply a superposition of its corresponding value and eye fixation, which is normalized to zero mean and unit standard deviation. This is calculated for each eye fixation, and subsequently the average NSS of all fixations results in a score that can be, (a) zero, when there is no link between eye fixation and face regions; (b) negative, when the fixations are on non-face regions; or (c) positive, when they are projected on the face regions. The higher the NSS values, the more are the face regions focused upon.

– **CC criterion:** Linear Correlation Coefficient measures the relationship between two images, widely used for evaluating the performance of visual attention models [Jos+05; LM+06; RBC06; Raj+08]. The linear correlation is the strength of a linear relationship between two variables:

$$CC(M^h, M^f) = \frac{\sum_{x,y}(M^h_{(x,y)} - \mu_{M^h}) \cdot (M^f_{(x,y)} - \mu_{M^f})}{\sqrt{\sigma^2_{M^h} \cdot \sigma^2_{M^f}}}$$

where M^h and M^f represent the eye fixation map and face map respectively. μ and σ^2 are the mean and variance of these maps. When the correlation CC is close to $+1/-1$ there is almost a perfectly linear relationship between the two variables.

– **KL criterion:** Symmetric Kullback Liebler divergence [HH99; GHV09] computes the gap, or similarity between two probability distributions using the equation:

$$KL(M^f, M^h) = \sum_i (M^f(i) - M^h(i)) \cdot log \frac{M^f(i)}{M^h(i)}$$

M^f : face map
M^h : eye fixation map

The closer the distributions the better are the results; they are always greater than or equal to zero.

Each of these different criteria have their advantages and disadvantages. Some are invariant to monotonic transformations, the AUC, the TC and the KL, growing predicted regions of interest has no impact on the scores. In other words, they are not sensible to increased false alarms, and are more robust. On the contrary, the NSS is not invariant, it takes all the fixations into account. This behavior helps the criterion to function correctly, when the the binary-classifier based criteria gives similar scores in cases with less and more false alarms.

Many different criteria are proposed in the literature to evaluate predicted regions against true values, real fixations. Some use binary classifiers (the AUC and the TC), others are referred to as dissimilarity scores (the NSS and the CC), with increasing scores representing good predictions. In contrast, the KL criterion is divergence, and has a reverse effect compared to all other criteria. All in all, most of them are quite simple to compute. In this study, we present most of these mentioned criteria. However, we

prefer to use AUC for significance tests due to its robustness, while the NSS to evaluate the strength of regions after modulation for its sensibility.

Fixation duration

Cognitive systems interact with the scene to determine where, and how long to fixate. The position of fixation points toward the region of interest, while its duration amounts for the attentional processing engaged to that location [JC76; Ray98; Hen07]. Faces being inherently relevant to primates, they provide important social and visual information [Guo+03; Guo+06], leading to long-lived fixations. Different studies have found several factors contributing to longevity of fixations on face features [Yar67; BBK08]. Some of these are intuitive such as social significance and expertise of faces [And98], whereas others are closely linked to eye movement patterns and cognitive demands [Ros+00]. In this study, analysis of fixation durations on scenes with faces can help to understand the temporal influence of different factors, such as eccentricity, area and number of faces.

3.3.3 Database

The video database comprised a variety of face content, such as scenes with cases of one or more faces at different locations, as well as different sizes. We labeled 14,600 frontal and upright faces in total for the entire video database (14,155 frames), to create a face ground truth for this study. We also labeled turned faces when the facial features such as eyes and mouth regions were distinguishable. Moreover, background faces with blurred features were ignored in favor of foreground faces. The distribution of the face ground truth for SM-I video database is shown in Figure 3.7, while Figure 3.8 shows some sample frames with hand-labeled face bounding boxes.

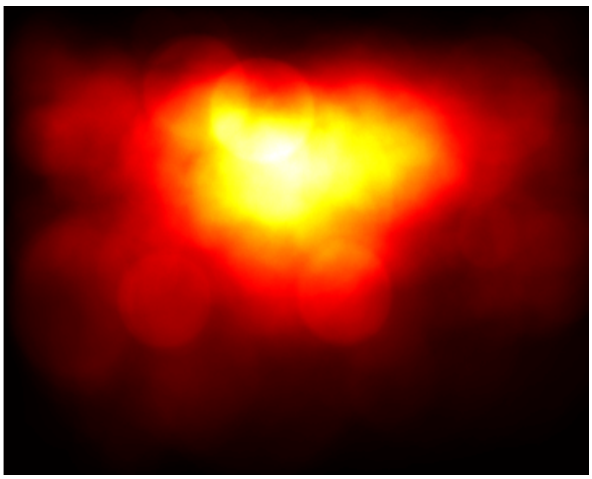


Figure 3.7: Representation of faces present in the entire video database. The surface map was created by placing a Gaussian over the face bounding boxes. The dimensions of the bounding box determine the variance of the 2D Gaussian from origin in horizontal and vertical axis, whereas the amplitude of the function was kept constant for all faces. All values outside the elliptical Gaussian face were set to zero.

During the experiment, the eye tracker recorded participants' eye movements at 500Hz—20 recordings for two eyes per frame and per participant. The recordings are then used to calculate corresponding fixations and saccades. In this study, we used these eye fixations to study different factors influencing the interest of faces in a dynamic scene.

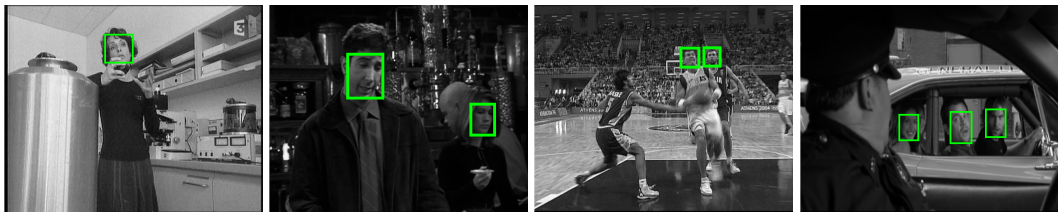
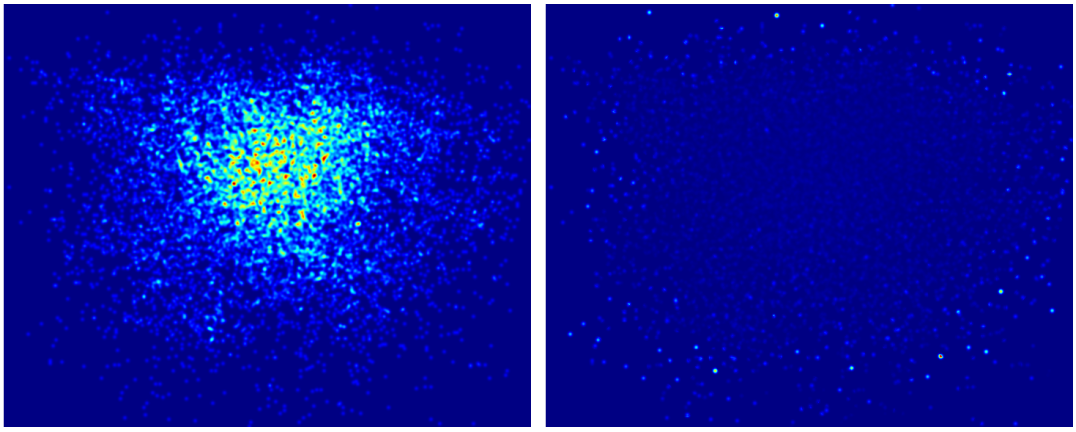


Figure 3.8: Sample video frames with hand-labeled faces from SM-I video database.

In total, 23,797 fixations were recorded for 14 participants, with 11,155 fixations on scene with at least one face. The number of fixations in different cases of number of faces in a scene are summarized inTable 3.1. Figure 3.9 shows the positions of all fixations for scenes with faces represented as landscapes. These were created by placing a Gaussian with a diameter of 2° of visual angle (equivalent to the fovea) centered at each fixation point, summed all the Gaussians together, and normalized the resulting sums. For the duration-weighted landscape, the height of each Gaussian was proportional to the duration of that fixation in milliseconds.

	One face	Two faces	More than two faces
No. of sample frames	3,335	2,317	1,151
Total Fixations	5,425	3,937	1,793

Table 3.1: Total number of fixations for one, two, or more faces in a scene. In the study, we consider fixations for scenes with one, or two faces.



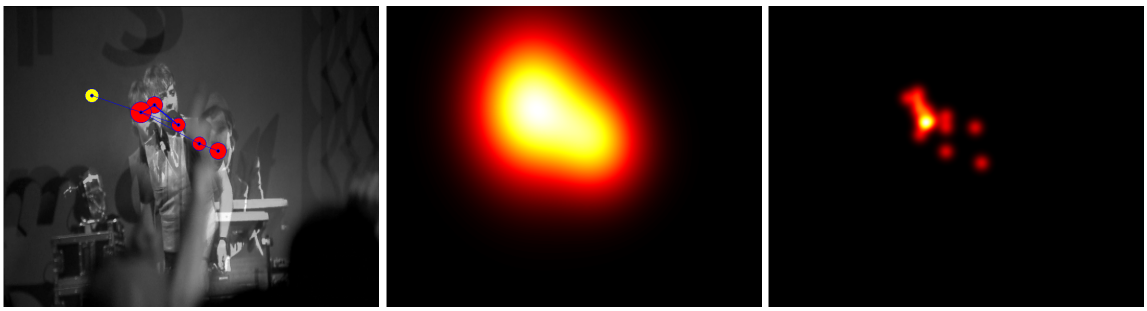
(a) Fixation distribution.

(b) Fixation distribution weighted by fixation durations.

Figure 3.9: Representation of the positions of all fixations for all participants in a scene. These surface maps were created by placing a Gaussian with a diameter of 2° of visual angle (equivalent to the fovea) centered at each fixation point, summing the Gaussians, and normalizing the height of the resulting sums. For the duration-weighted surface map, the height of each Gaussian was proportional to the duration of that fixation in milliseconds.

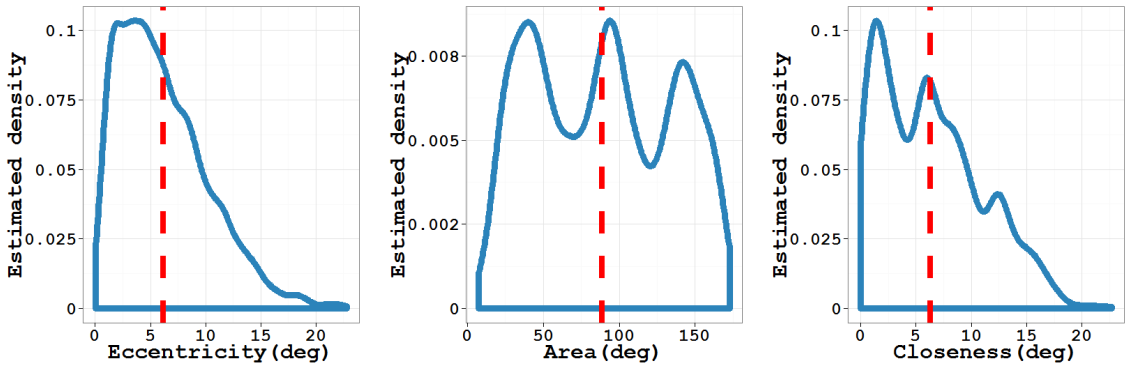
3.3.4 Statistical analysis

To measure the influence of *'number of faces'* in a scene, we compute 2 (one or two faces) ANOVA for all 191 video snippets. This is done for all evaluation measures. Likewise, to determine the influence of eccentricity, area and closeness between two faces, the AUC comparison criterion was averaged across subjects for 111 video snippets in the case of one face, whereas 80 video snippets in the case of two faces. Then, significance tests for main and interaction effects of different influencing factors were computed using linear regression. Here, linear regression was used because the predictor variables, or influencing factors were continuous. Figure 3.11 shows the density estimates of these variables after applying a kernel smoothing method to the distributions. In the study, for clarity, we only present statistics using the AUC criterion, since in most of the case the comparison criteria (AUC, TC, NSS, CC and KL) provide the same conclusion.



(a) (b) (c)

Figure 3.10: From left to right (a) scanpath for one participant during viewing of a dynamic scene (illustrated as superimposed first, middle and last frames). The circles are the fixations connected by preceding saccades, while the diameter of the circles corresponds to fixation duration ($40ms$ = $0.2°$). The fixation on scene onset is illustrated in yellow, whereas the following fixations are red. We observe that the entire scanpath remains on the objects of interest in the scene, the faces. (b) A heat map representing the Gaussian face regions for the entire scene, and (c) a heat map with 'second' eye fixation with Gaussian ($\sigma = 0.5°$) for all fifteen participants.



(a) Eccentricity of face from fixation.

(b) Surface area of face.

(c) Closeness between two faces.

Figure 3.11: Density estimates of different variables using a kernel smoothing method.

3.4 Results: Interest of faces

Faces are unique in the sense that they can hold attention more in natural scenes compared to other object stimuli [Guo+06; FW+08]. Many studies find evidence of different perceptual mechanisms specialized in face processing [Mec+04; KY06; Sum+06]. Others explore the preferential advantage of face in the human visual system [JM01; Tau09], and agree that they are processed as early as other low-level features through some high-level holistic processing mechanisms [HH05; HH06]. In this study, we observe the interest in faces using dispersion of fixations among participants and distance of fixations from face.

3.4.1 Inter-observer congruency

Gaze is highly influenced by faces compared to other object stimuli. We analyzed inter-observer congruency, or dispersion among participants' fixations to observe this interest of faces in scenes.

IOC averaged across subjects for 191 video snippets with faces (111 with one face and 80 with two faces) is shown as a function of fixation number in Figure 3.12. A 2 (one or two faces) ANOVA revealed significant main effects of number of faces ($F_{(1,189)} = 4.588$, $p = 0.033, \eta_p^2 = 0.024$). Paired samples t-tests (2-tailed) at each fixation showed significant differences between dispersion in the two cases (one and two faces) for second, third and fourth fixations, $t_{(169)} = -2.592$, $p = 0.01$, $t_{(171)} = -2.167$, $p = 0.03$, and $t_{(162)} = -2.328$, $p = 0.02$ respectively.

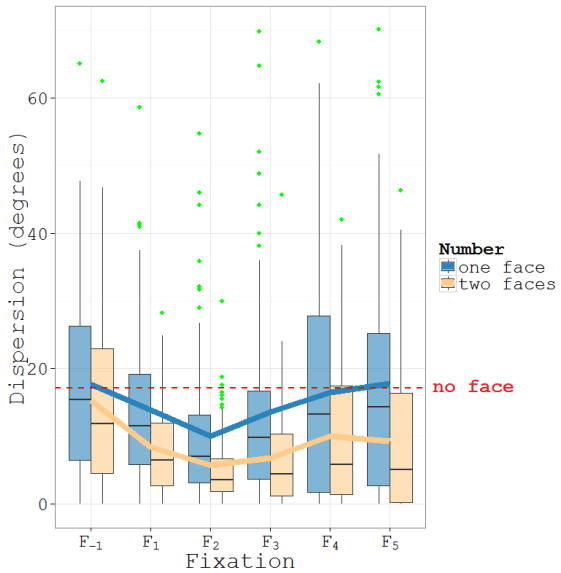


Figure 3.12: Inter-observer congruency (IOC), or fixation dispersion among participants for one, two faces and no faces. We took first five fixations $\{F_1, F_2, F_3, F_4, F_5\}$ after current scene onset and fixation F_{-1} from the previous clip snippet (fixation just before onset of current scene). The solid lines are the mean dispersions over time for one and two faces, while the red dashed line is the mean dispersion for scenes with no faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

Along time, IOC among participants was smaller for fixations on scene onset, followed by fixations that either converged to a face, when it is the only region of interest, or bifurcated in the direction of the two competing faces. The later scenario resulted in large dispersion

among participants' fixations. As the scene progressed and exploration done, dispersion became similar for one or two faces.

3.4.2 Minimum fixation distance

Minimum fixation distance, or distance of participant's fovea, to face highlights the preference of face stimuli in a scene. We measured the distance of face to the fixation in the case of one face, and distance to the face with minimum eccentricity from the fovea in the case of two faces. The distance is computed between the fixation coordinates and the side of face.

Distance averaged across subjects for 191 video snippets with faces (111 with one face and 80 with two faces) is shown as a function of fixation number in Figure 3.13. Paired samples t-tests (2-tailed) at each fixation showed significant differences between distances in the two cases (one and two faces) only for the fifth fixation, $t_{(64)} = 2.733$, $p = 0.008$.

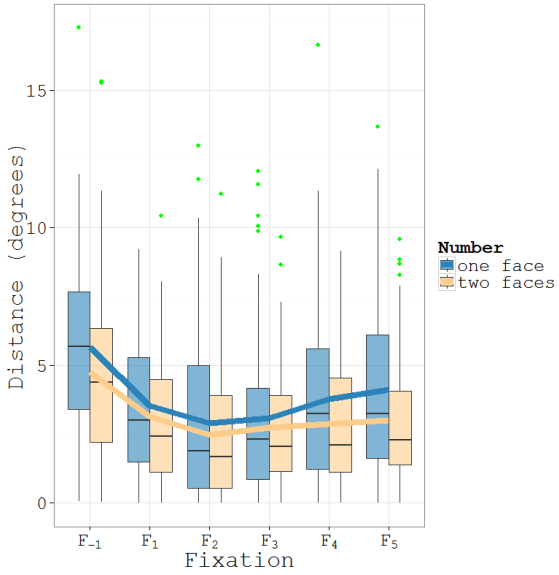
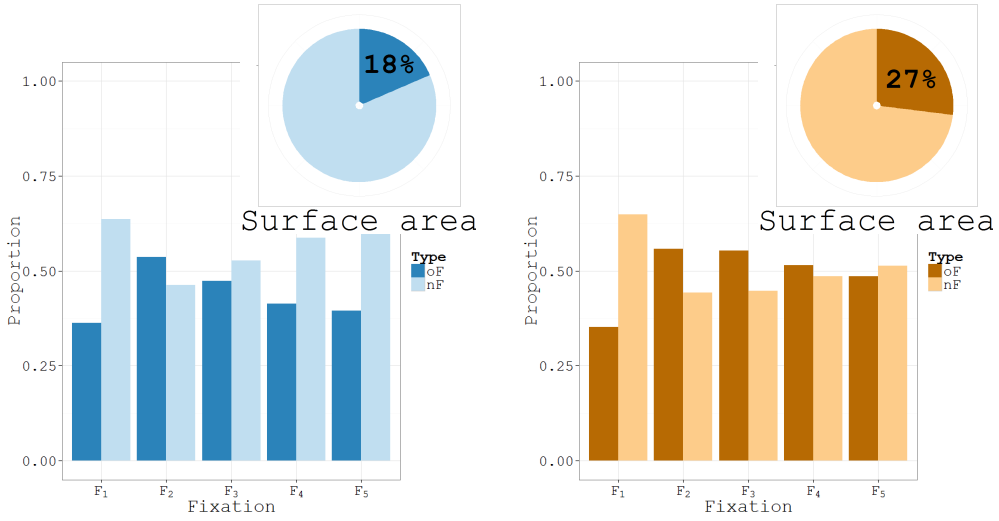


Figure 3.13: Minimum fixation distance, or distance of the fovea of participant, to face in the case of one face, while to face with minimum eccentricity from fovea in the case of two faces. The distance is computed between the fixation coordinates and the side of the face. We took first five fixations $\{F_1, F_2, F_3, F_4, F_5\}$ after current scene onset and fixation F_{-1} from the previous clip snippet (fixation just before onset of current scene). The solid lines are the mean distances over time for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

3.4.3 Fixation proportion

Initially, the proportion of fixations made on face regions in a scene are smaller, which is attributed to a center-looking strategy on scene onset. As the scene progresses, at the second fixation F_2, the proportion of fixations landing within the face regions becomes more than 50% of the total fixations. This large proportion remains for the following fixations, true in the case of both one and two faces (Figures 3.14a and 3.14b). The preference of fixating faces becomes more prominent with the observation that the surface area occupied by face regions is much smaller compared to the rest of the visual stimuli. However, they comprise a significant proportion of the total fixations for scenes with faces.



(a) One face (b) Two faces

Figure 3.14: Proportion of fixations made on or off face regions, denoted as *'on-face' (oF)* and *'not-on-face' (nF)* fixations respectively. The pie chart represents the surface area of the face ellipses f_i, computed as $\sum_i \pi ab$ and $\pi(15° \times 10°) - \sum_i \pi ab$ for oF and nF regions. We took first five fixations $\{F_1, F_2, F_3, F_4, F_5\}$ after current scene onset.

We conclude that faces, when present in a complex scene, are considered potential regions of interest. We found that participants remained in proximity to face along time. In addition to the inherent interest of faces, its eccentricity and and area are also important. Since these two factors could lead to less distorted information to form a recognizable object. Consequently, resulting in fixations made much closer to a face in a scene.

3.5 Results: Faces and comparison criteria

Faces in a scene are certainly one of the regions of interest in dynamic stimuli, and they do influence eye movements of the participants. In this section, we used the comparison criterion, the AUC, to evaluate the face maps M^f against the eye fixation maps M^h. The aim was to analyze the spatial importance of face regions. The quality of the maps corresponding to these regions is determined using some comparison criterion.

3.5.1 Number of faces

AUC scores averaged across subjects for 191 video snippets with faces is shown as a function of fixation number in Figure 3.26a. We observe that faces represent regions of interest, and they are frequently attended. However, their influence degrade when two faces are presented.

A 2 (one or two faces) ANOVA revealed significant main effects of number of faces ($F_{(1,189)} = 27.05$, $p < 0.001, \eta_p^2 = 0.14$). Paired samples t-tests (2-tailed) at each fixation $\{F_1, F_2, F_3, F_4, F_5\}$ showed significant differences between scores in the two cases (one and two faces) for all five fixations except the first fixation, $t_{(78)} = 1.69$, $p = 0.095$, $t_{(77)} = 2.982$, $p = 0.003$, $t_{(78)} = 3.344$, $p < 0.001$, $t_{(74)} = 2.207$, $p = 0.03$ and $t_{(64)} = 2.571$, $p = 0.012$.

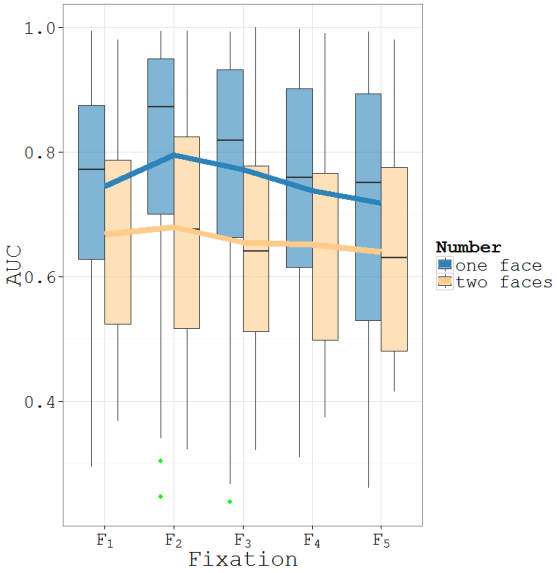


Figure 3.15: Temporal evolution of scores for AUC comparison criterion for *'number of faces'*—one and two faces. We took first five fixations $\{F_1, F_2, F_3, F_4, F_5\}$ after current scene onset. The solid lines are the mean scores over time for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

3.5.2 Eccentricity of faces

AUC scores averaged across subjects for 191 video snippets (111 with one face and 80 with two faces) is shown as a function of eccentricity in Figure 3.27a. We observe that the performance of faces decreased with increasing eccentricity in both cases (one and two faces), comparatively lower scores in the latter case. However, the drop from fovea to periphery for one face was more apparent.

In the case of one face, a significance test using linear regression revealed significant main effects of eccentricity of face ($F_{(1,109)} = 29.92$, $p < 0.001, \eta_p^2 = 0.27$). In the case of two faces, a significance test for eccentricity of face (one closest to fovea, or previous fixation) using linear regression revealed difference in AUC scores to be insignificant.

Paired samples t-tests (2-tailed) on each category showed significant differences between comparison criterion scores for the two cases (one and two faces). To do this, faces were categorized using three regions around fovea (previous fixation): $0° - 2°$ fovea, $2° - 7°$ parafovea and above $7°$ is periphery. We found significant differences in all regions, $t_{(59)} = 3.41$, $p = 0.001$, $t_{(181)} = 4.567$, $p < 0.001$ and $t_{(135)} = 2.87$, $p = 0.005$.

3.5.3 Area of faces

AUC scores averaged across subjects for 191 video snippets (111 with one face and 80 with two faces) is shown as a function of face area in Figure 3.28a. We observe that faces with larger surface areas attract more gaze compared to ones with smaller surface areas.

In the case of one face, a significance test using linear regression revealed significant main effects of area of face ($F_{(1,109)} = 54.07$, $p < 0.001, \eta_p^2 = 0.5$), and a significant interaction effect between eccentricity and area of face ($F_{(3,107)} = 29$, $p < 0.001, \eta_p^2 = 0.81$). In the case of two faces, a significance test for area of face (one closest to fovea, or previous fixation) using linear regression revealed difference in AUC scores to be insignificant.

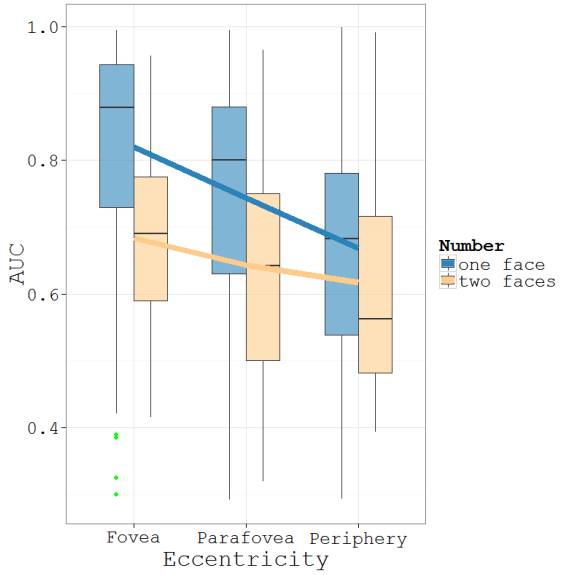


Figure 3.16: Scores for AUC comparison criterion as a function of *'eccentricity from fovea'* for one and two faces. Faces were categorized using three regions around fovea (previous fixation): $0° - 2°$ fovea, $2° - 7°$ parafovea and above $7°$ is periphery. In both cases, we took at most five fixations (first five after scene onset). The solid lines are the mean scores for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

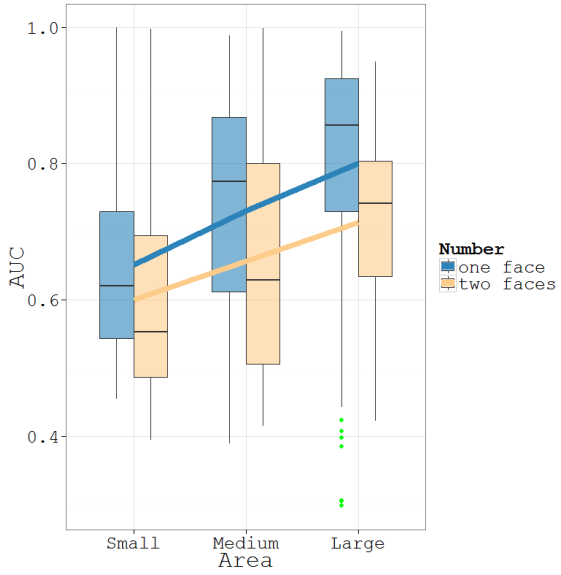


Figure 3.17: Scores for AUC comparison criterion as a function of *'face area'* for one and two faces. Faces were categorized into three equally distributed area categories: $0^{(\circ)^2} - 50^{(\circ)^2}$ small, $50^{(\circ)^2} - 75^{(\circ)^2}$ medium and above $75^{(\circ)^2}$ large faces. The area of the face at minimum eccentricity from fovea was considered in the case of two faces. In both cases, we took at most five fixations (first five after scene onset). The solid lines are the mean scores for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

Paired samples t-tests (2-tailed) on each category showed significant differences between

comparison criterion scores for the two cases (one and two faces). To do this, faces were categorized into three equally distributed area categories: $0^{(\circ)^2} - 50^{(\circ)^2}$ small, $50^{(\circ)^2} - 75^{(\circ)^2}$ medium and above $75^{(\circ)^2}$ large faces. We found significant differences when face fixated was medium or large-sized, $t_{(122)} = 3.672$, $p < 0.001$ and $t_{(150)} = 5.06$, $p < 0.001$ respectively.

3.5.4 Closeness between two faces

In addition to evaluating the three main face influencing factors (number, eccentricity and area of faces), in the case of two faces, we analyze closeness between the competing faces. AUC scores averaged across subjects for 80 video snippets with two faces is shown as a function of closeness between faces in Figure 3.18. We observe that closer face regions result in higher scores compared to faces farther.

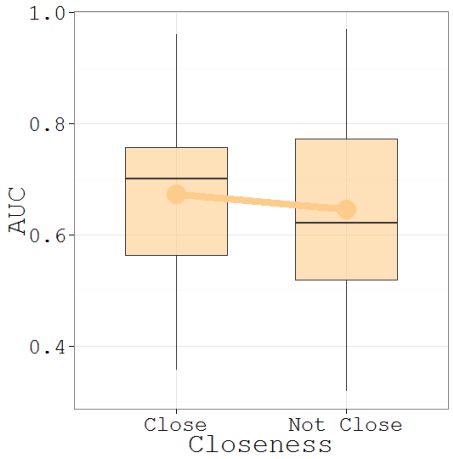


Figure 3.18: Scores for AUC comparison criterion as a function of *"closeness"* between two faces. Faces were categorized into two categories: 'Close' and 'Not Close'. To categorize we used the notion of critical spacing that is proportional to eccentricity. It is defined by Bouma's proportionality constant [Bou70], which is an object at an eccentricity E might be crowded by other objects as much as $0.5E$ away. Here, we took at most five fixations (first five after scene onset). The solid line is the mean score two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

In the case of two faces (80 video snippets), a significance test for closeness as a function of eccentricity and area of face (one closest to fovea, or previous fixation) using multiple linear regression revealed significant main effects of closeness ($F_{(1,78)} = 8.05$, $p < 0.001, \eta_p^2 = 0.10$), and a significant interaction effect between (i) eccentricity and closeness of face ($F_{(3,76)} = 3.502$, $p = 0.008, \eta_p^2 = 0.14$) (ii) area and closeness of face ($F_{(3,76)} = 6.744$, $p = 0.002, \eta_p^2 = 0.27$), and (iii) eccentricity, area and closeness of face ($F_{(7,72)} = 4.755$, $p = 0.009, \eta_p^2 = 0.46$). We conclude that there is an influence of closeness. The influence increases with increasing eccentricity, on the contrary, decreases with increasing area.

3.5.5 Interpretation

The performance of face regions is high, but it drops when there are multiple faces to fixate in a scene. The significance tests confirm that there are differences in comparison criteria for one or two faces over time. As the scores represents the spatial performance of face regions, they decrease with increasing load.

Faces in foveal regions perform better even when there is competition; that is, the comparison criterion scores are similar for both one- and two-face cases. In the parafovea, faces are equally distinguishable only in the absence of competing faces. Lastly, in the periphery, both one- and two- face conditions show lowest performance due to the under-representation of information in periphery.

Size of face also effects the interest of faces, and increase in size can compensate for the lost visual resolution in periphery. Moreover, the closer are the two competing face, more is their spatial strength to attract gaze. Hence, limiting the effects of competition.

In summary, we found significant effects all influencing factors on the interest of faces. Number and eccentricity of faces limiting their interest. In contrast, area and closeness reduce the aforementioned influences.

3.6 Results: Faces and fixation duration

The configuration, two small dark regions and one large dark region, carries extremely effective facial information for the human visual system. To extract this information, faces are fixated more often and for longer durations compared to other object stimuli in natural scenes. In this section, we analyzed fixation duration with respect to the different influencing factors: eccentricity, area and number of faces. The aim was to analyze the amount of attention allocated to the face regions.

3.6.1 Number of faces

Fixation durations averaged across subjects for 191 video snippets (111 with one face and 80 with two faces) with faces is shown as a function of fixation number in Figure 3.19. We observe that durations for video snippets with faces are higher compared to video snippets with no faces. Along time, the first fixation is shorter followed by longer second fixation. In the case of one face, the second fixation is the longest, followed by more fixations comparatively longer than those in the case of two faces.

A 2 (one or two faces) ANOVA revealed marginally significant main effects of number of faces ($F_{(1,189)} = 3.306$, $p = 0.07, \eta_p^2 = 0.017$). Paired samples t-tests (2-tailed) at each fixation $\{F_1, F_2, F_3, F_4, F_5\}$ showed insignificant differences between durations in the two cases of number of faces (one and two faces) for all five fixations.

3.6.2 Eccentricity of face

Fixation duration averaged across subjects for 191 video snippets (111 with one face and 80 with two faces) is shown as a function of eccentricity in Figure 3.20. We observe that the duration of fixations were longer in the case of one face when the face is located within the participants foveal region. In parafovea region, the fixation durations became similar. However, there was a difference again in the durations in peripheral region; interestingly, shorter for one face compared to two faces.

In the case of one face, a significance test using linear regression revealed significant main effects of eccentricity of face ($F_{(1,109)} = 6.55$, $p = 0.01, \eta_p^2 = 0.06$). In the case of two faces, a significance test for eccentricity of face (one closest to fovea, or previous fixation) using linear regression revealed difference in fixation duration to be insignificant.

Paired samples t-tests (2-tailed) on each category showed marginally significant differences between durations for the two cases (one and two faces). To do this, faces

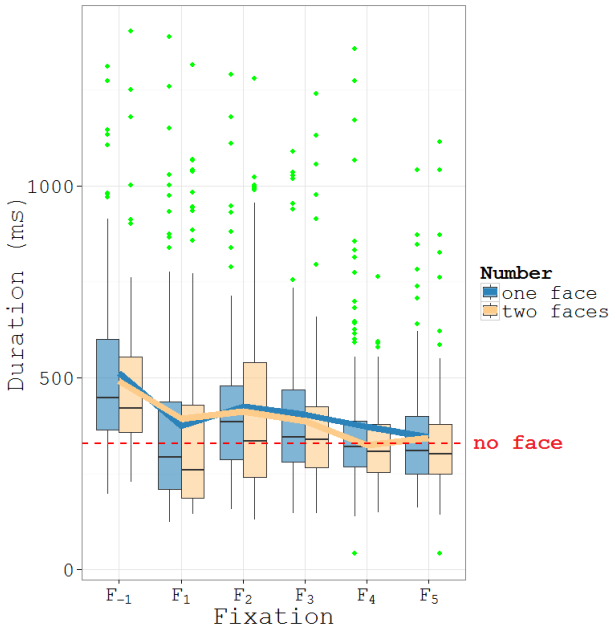


Figure 3.19: Fixation duration for one and two faces. We took first five fixations $\{F_1, F_2, F_3, F_4, F_5\}$ after current scene onset and fixation F_{-1} from the previous scene (fixation just before onset of current scene). The solid lines are the mean durations over time for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

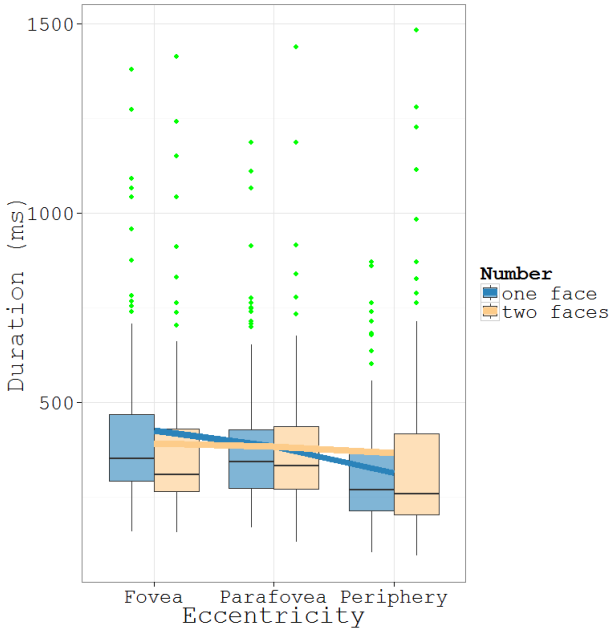


Figure 3.20: Fixation duration as a function of eccentricity of face (one and two faces) from fovea, or fixation. The faces were categorized using three regions around fovea (previous fixation): $0° - 3°$ fovea, $3° - 7°$ para-fovea and above $7°$ is periphery. In both cases, we took at most five fixations (first five after scene onset). The solid lines are the mean scores for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

were categorized using three regions around fovea: $0° - 2°$ fovea, $2° - 7°$ parafovea and above $7°$ is periphery. We found significant differences in duration when face fixated is fovea, $t_{(59)} = 1.87$, $p = 0.066$.

3.6.3 Area of face

Fixation duration averaged across subjects for 191 video snippets (111 with one face and 80 with two faces) is shown as a function of face area in Figure 3.21. We observe that large faces are fixated for longer durations compared to smaller sized faces.

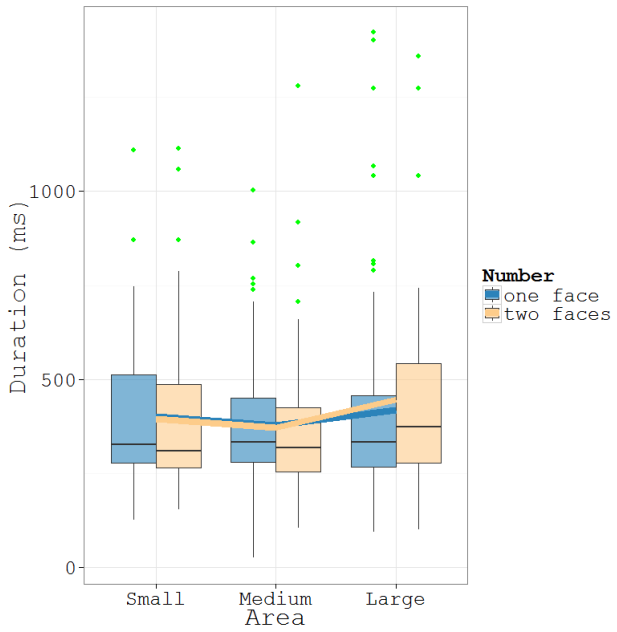


Figure 3.21: Fixation duration as a function of face area for one and two faces. Faces were categorized into three equally distributed area categories: $0^{(o)^2} - 50^{(o)^2}$ small, $50^{(o)^2} - 100^{(o)^2}$ medium and above $100^{(o)^2}$ large faces. The area of the face at minimum eccentricity from fovea was considered in the case of two faces. The error bars represent the mean standard error. In both cases, we took at most five fixations (first five after scene onset). The solid lines are the mean scores for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

In the case of one face, a significance test using linear regression revealed significant interaction effect between eccentricity and area of face ($F_{(3,107)} = 14.71$, $p < 0.001$, $\eta_p^2 = 0.41$). In the case of two faces, a significance test for area of face (one closest to fovea, or previous fixation) using linear regression revealed difference in fixation durations to be insignificant.

Paired samples t-tests (2-tailed) on each category showed significant differences between durations for the two cases (one and two faces). To do this, faces were categorized into three equally distributed categories: $0^{(o)^2} - 50^{(o)^2}$ small, $50^{(o)^2} - 75^{(o)^2}$ medium and above $75^{(o)^2}$ large faces. We found significant differences in fixation durations only when faces fixated were large-sized, $t_{(150)} = -2.055$, $p = 0.04$.

We conclude that fixation durations for different sized faces are similar, except the durations are considerably longer for one large face. It is apparently due to the large face representing the only region of interest in the entire scene.

3.6.4 Closeness between two faces

Fixation durations averaged across subjects for 80 video snippets with two faces is shown as a function of closeness between faces in Figure 3.22. We observe that closer face regions introduce two equally competing regions of interest leading to shorter fixations. In contrast, two faces far apart leads to no competition between the two regions. In this case, the interest of faces is determined by their eccentricity and area, or similar to presenting one face resulting in longer fixations.

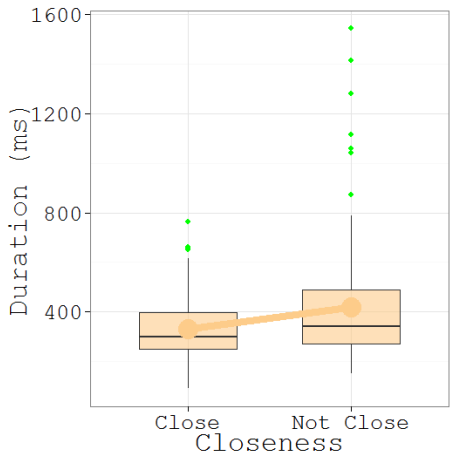


Figure 3.22: Fixation duration as a function of *'closeness'* between two faces. Faces were categorized into two categories: 'Close' and 'Not Close'. To categorize we used the notion of critical spacing that is proportional to eccentricity. It is defined by Bouma's proportionality constant [Bou70], which is an object at an eccentricity E might be crowded by other objects as much as $0.5E$ away. Here, we took at most five fixations (first five after scene onset). The solid line is the mean duration for two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

In the case of two faces (80 video snippets in total), a significance test for closeness as a function of eccentricity and area of face (one closest to fovea, or previous fixation) using multiple linear regression revealed significant interaction effect between (i) eccentricity and closeness of face ($F_{(3,76)} = 3.26$, $p = 0.026, \eta_p^2 = 0.03$), and (ii) eccentricity, area and closeness of face ($F_{(7,72)} = 3.191$, $p = 0.005, \eta_p^2 = 0.31$).

We conclude that there is an influence of closeness between faces on fixation durations. Closer are the faces, more is a competition between the two regions. However, in the absence of the influence, interest of the regions is resolved based on other influencing factors.

3.6.5 Interpretation

On scene onset, the participants follow a center-looking strategy where the center of the face region is first fixated. This first fixation is short, followed be long fixations to extract maximum of information in the visual field. In the case of one face, fixations are longer compared to when there are two faces. One face in a scene leads to higher proportion of initial fixations that decreases as the scene exploration progresses. The only face—in the absence of competition—is the potential region of attention, and it leads to longer and fewer fixations to extract the visual information.

Faces are informative, and they require long fixations. Without competition, scene with only one face, leads to longer fixations when the face is fixated. On the contrary, fixations

outside the face regions are similar irrespective of the number of faces. Furthermore, the inherent importance of faces to the visual system is observed by the high proportions of fixations made in proximity to face regions. This result also gives a hint of many subtle eye movements made to extract maximum from the information rich faces.

Large faces potentially carry more information with well-represented facial features. We found that the duration of fixations are similar with no significant differences across different face sizes, with or without competition, except large faces. Large faces when presented alone led to longer fixations compared to ones with some competition.

Closeness between two faces leads to shorter fixations, linked to competition. However, when farther from one another, leads to longer fixations due to the face at smaller eccentricity treated as the only region of interest. On the contrary, the evaluation of faces using comparison criterion suggests that closeness increases the interest of the face regions.

In summary, we found significant influences of eccentricity of face, unless there is no competing faces. In case of competition, the influence reached significance after interactions with the factor of closeness between faces. This observation shows that closeness makes the competition for attentional resources more tougher, and hence the duration of fixations made is similar to average duration. Size of face was not found to influence duration in any case, with or without competing faces. However, the interactions of eccentricity and size of face reached significance. In contrast to comparison criterion, the factor of closeness between faces has no main effects, whereas the interaction factors with the influence of eccentricity and size of faces show significant differences in fixation durations.

3.7 Discussion

Over the years, a number of studies have been conducted regarding the influence of faces on gaze, but mostly using static stimuli. However, it has rarely been reported for dynamic stimuli (in videos). In videos, the object stimuli can be correlated over different times and positions, as the stimuli does not change randomly over space and time. This regular presentation of visual stimuli is quite efficient to represent and transmit information [DA95]. It is particularly true in the case of faces where dynamic stimuli can offer more information compared to static stimuli [BPM07].

Different studies using static stimuli show that participants tend to fixate faces based on saliency [BBK09], or due to their social importance [BBK08]. Since the study involves an experiment with free-viewing participants, we imply that the longer and frequent fixations are closely linked to the organization of contours that make up a recognizable object—the case of faces [FW+08]. In contrast, during social interactions mouth is fixated [LM03], whereas determining emotional state requires holistic face processing [CJT11]. All these strategies are dynamically executed depending on the type of visual information required. In short, the amount of information that the face has to offer determines the gaze patterns [McC+88].

Faces are difficult to ignore in a complex visual scenes [LRR03], due to their light-shadow pattern, with eyes as two small dark regions and mouth as a large dark region. The configuration carries extremely effective facial information for the human visual system [JHC92], which tends to make many subtle eye movements on the face [Vel02; FFLM11]. Initially, they are fixated at the center [HC08], whereas the next move is linked to two main factors. First, the type of information sought to complete specific goals, for example to determine human gaze [HSP12], identity [HJ01; Kna03] or emotions [ASC05]. Second, the conditions of social interaction, for example to facilitate face-to-face communication by increasing intelligibility of speech [BTD00; LM03; BPM07; Vo+12]. Since, we use dynamic

stimuli without sound, there is no impact of auditory cues on visual information extracted. We can imply that the presentation of visual stimuli alone leads to the centralization of gaze, resulting in longer fixations to extract maximum information from faces.

Unlike previous studies, we used a video database that comprised face content with complex backgrounds and varying face luminance and orientations. In fact, the database was compiled to understand the early visual attention mechanisms, and to evaluate the prediction of a visual saliency model. The varying face content makes the video database useful to study faces in real videos. Consequently, the observations can help to incorporate 'face features' into visual saliency models, increasing their predictability.

Since little of the face content used comes from famous movies, less than 5% in proportion, the impact of familiar faces was negligible. Also, in this study, we investigated the impact of eccentricity and number of faces in early vision with minimal involvement of higher perceptual processing. Therefore, we infer that familiar faces have no influence.

The performance of visual stimuli drop as they are presented farther away in the periphery. However, in the case of faces, they are preferred even at large eccentricities. The observation is coherent with findings of different studies that the influence of faces become more apparent with increasing eccentricities [Tho+01; TJC09]. They can be fixated even when presented in the periphery with limited spatial information [GLR11]. On the other hand, in the case of two faces the drop more compared to one face. In a related study [GLR11] when a face is presented alongside similar faces, the performance to fixate a face drops, possibly a consequence of the limited information-processing capacity of the peripheral vision. Moreover, the competing faces probably influence each other based on their locations, determining the attentional resources allocated for the faces. In this study, we observe the effects of eccentricity on faces, more in the case of two faces. In the some cases, it resolves the competition between two faces. Area of face can limits the effects of eccentricity, and it can also reduce the crowding of facial information increasing the influence of faces. In conclusion, it is important to understand the interest of faces with respect to these different influencing factors.

For the time being, it can be stated that face does attract attention, and it becomes more apparent when multiple face stimuli are competing for limited attentional resources. Based on the previous studies, the findings are coherent. They could be helpful to understand the biases influencing faces, which in turn would support the possibility of adding a modulated face pathway to a recently proposed visual saliency model in [RPH12], to predict eye movements. Ultimately, the modulation could prove to be crucial for the implications of the model, for instance used in a social robot. Nevertheless, further investigation is required to understand the behavior of subtle fixations made on different face regions, and the impact of competing face stimuli on fixation duration and saccade amplitudes.

3.8 Proposed modulation of face pathway

Based on the findings from this study, we propose a simple modulation for faces. We modulate three cases of faces in scenes: one face, two faces, and more than two faces. In the case of one face, the initial weights remain unchanged, since we assume that they are preferred even at larger eccentricities. In the case of two faces, we modulate the weights of the faces based on influences of eccentricity, area and closeness. In the case of more than two faces, we select two faces among many based on their eccentricity—faces with smaller eccentricities. The weights for these selected faces are computed using the same method as one used for two faces. The rest of the faces in this case, more than two faces, are set to one-half of the face with lower modulation weight.

To start computing modulation weights for the faces, we follow two steps. First, we initialize the confidence c for all the hand-labeled faces to 1.0. This initial weight represents the maximum amplitude of the Gaussian function applied to the faces. Second, we defined the different influencing factors for individual faces (i) Eccentricity E is the euclidean distance between the origin of the face ellipse (O_x, O_y) and the fixation (C_x, C_y), (ii) Area A is the surface area of the face ellipse, and (iii) Closeness C is a coefficient that modulates the influence of eccentricity based on closeness between two faces. All these factors are detailed in Section 3.3.1.

Consider two faces (f^1, f^2) with the three influencing factors E, A and C. The weights for the faces (f^1, f^2) are computed using the following equations with k set to 6 (Algorithm 1).

The weight of eccentricity of face E from fixation is computed using:

$$\omega_E = \frac{2}{1 + e^{-(-k \cdot E')}}$$

The weight based on area of face A is computed using:

$$\omega_A = \frac{1}{1 + e^{-k \cdot A'}}$$

The weight based on the influencing factor of closeness C for two faces (f^1, f^2) is computed using:

$$\omega_C = \begin{cases} s, 2 \cdot s \cdot (1 - s) & \text{if } (E^{f^1} < E^{f^2}) \\ 2 \cdot s \cdot (1 - s), s & \text{otherwise} \end{cases}$$

where

$$s = \frac{1}{1 + e^{-k \cdot C'}}$$

The combined weight of all the influencing factors determine to the final modulation weights for faces (f^1, f^2):

$$\omega = \frac{1}{2.5} \cdot (\omega_E + \omega_A + \omega_C)$$

The weights are then used to modulate the face confidence c, which in this case is set to initial value 1.0.

$$c = \begin{cases} w \cdot c & \text{if } (f^1, f^2) \\ 0.5 \cdot arg\,min_w \cdot c & \text{otherwise} \end{cases}$$

Figure 3.23 illustrates the evolutions of weights (ω_E, ω_A and ω_C) as a function of increasing influencing of the three factors, eccentricity, area and closeness of face. The amplitudes of the Gaussians for the face maps are equal to the confidence c after modulation.

The resulting modulated face maps $M^{f'}$ are evaluated against the eye fixation maps M^h density maps using different comparison criteria. The mean NSS and CC scores in Table 3.2 show a 5% improvement in the predictability of face pathway after incorporating the influences of number of faces, eccentricity of faces, area of faces, and closeness between faces. The temporal evolution of NSS in Figure 3.24 shows an improvement along time. On the other hand, the mean AUC and TC remain similar indicating that the modulation did not effect the interest of faces, or the predicted regions of interest.

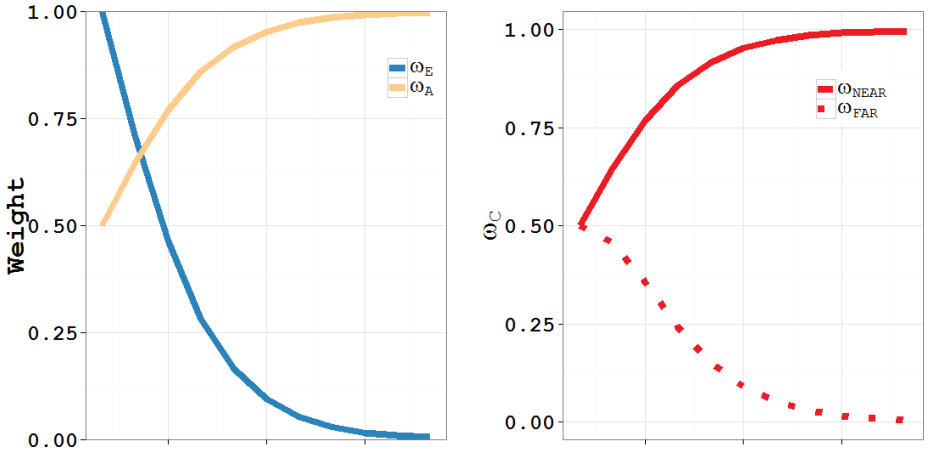
Algorithm 1 Calculate modulation weights for the faces

Precondition: Eccentricity E, Area A, and Closeness C between faces f^1, f^2. Set k to 6.

$\omega_E \Leftarrow 2/(1+\exp(-(-k \times E)))$
$\omega_A \Leftarrow 1/(1+\exp(-k \times A)))$

$s \Leftarrow 1/(1+\exp(-k \times C))$
if $E^{f^1} \leq E^{f^2}$ **then**
    $\omega_C \Leftarrow s$
else
    $\omega_C \Leftarrow s \times (1-s)$

$\omega \Leftarrow (\omega_E + \omega_A + \omega_C)/2.5$



(a) ω_E for eccentricity of face E and ω_A for area of face A.

(b) ω_C for closeness between two faces: ω_{NEAR} for face at smaller eccentricity, otherwise ω_{FAR}.

Figure 3.23: Illustration of modulation weights as a function of the three influencing factors.

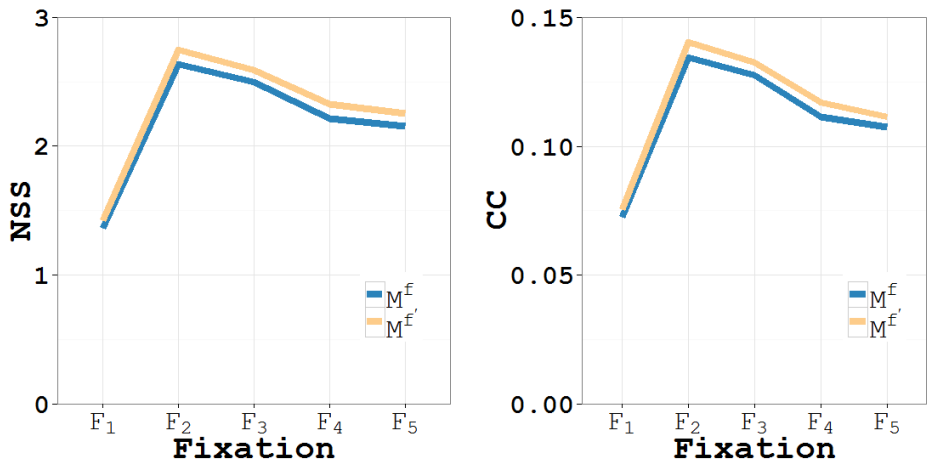
A visual saliency model predicts salient regions that are likely to be fixated by the participants. Hence, we use the distance of the location of face on the screen to the center of screen (G_x, G_y) for the modulation, instead of the eccentricity of face from fixation.

The modulated face maps $M^{f'}$ are evaluated against eye position density maps[2] using different comparison criteria. The mean NSS and CC scores in Table 3.3 show a 5% improvement in predictability of the face pathway after incorporating the influences of number, location, area and closeness of faces. The temporal evolution of NSS in Figure 3.25 shows an improvement along time. On the other hand, mean AUC and TC remain similar indicating that the modulation did not effect the interest of faces.

[2]For each frame, median position of the recordings for each participant was considered, called an eye position. A 2D Gaussian was added to each eye position with standard deviation equal to 1.0°. Finally, the processed eye positions for all participants for each frame are combined to obtain a human eye position density map

		M^f	$M^{f'}$
AUC	$\bar{x}$	0.66	0.66
	$SE_{\bar{x}}$	0.033	0.033
TC (%)	$\bar{x}$	35	35
	$SE_{\bar{x}}$	6.601	6.599
NSS	$\bar{x}$	2.17	2.27
	$SE_{\bar{x}}$	0.492	0.514
CC	$\bar{x}$	0.111	0.115
	$SE_{\bar{x}}$	0.024	0.025
KL (deg)	$\bar{x}$	0.96	0.96
	$SE_{\bar{x}}$	0.039	0.039

Table 3.2: Comparison criteria for simple face maps M^f and modulated face maps $M^{f'}$ using face data from ground truth evaluated against eye fixation maps.



(a) NSS evolution. (b) CC evolution.

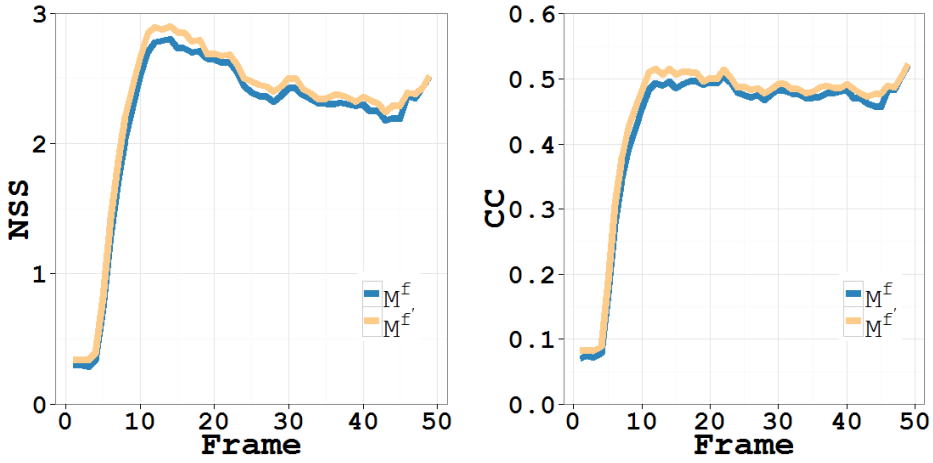
Figure 3.24: Evolution of comparison criteria for simple face maps M^f and modulated face maps $M^{f'}$ using face data from ground truth evaluated against eye fixation maps.

3.9 Conclusion

In conclusion, faces are fixated for long durations in a dynamic scene since it offers more visual information for perception. The influence of faces decreases with increasing eccentricity; in particular, the influence of eccentricity is more apparent for two competing faces. The initial fixation on scene onset is shorter compared to the following fixations. Furthermore, the high proportion of foveal fixations show that faces in a scene trigger fixations preceded by small saccades, essentially motivated to perform detailed analysis of the facial features. This trend of making fixations is much stronger for one face with longer fixations compared to two faces in a scene. The study is important to understand eye movements for complex object categories like faces in videos, in consequence their inclusion

		M^f	$M^{f'}$
AUC	$\bar{x}$	0.74	0.74
	$SE_{\bar{x}}$	0.010	0.010
TC (%)	$\bar{x}$	54	55
	$SE_{\bar{x}}$	2.015	2.008
NSS	$\bar{x}$	2.20	2.29
	$SE_{\bar{x}}$	0.094	0.091
CC	$\bar{x}$	0.43	0.45
	$SE_{\bar{x}}$	0.017	0.016
KL (deg)	$\bar{x}$	0.61	0.59
	$SE_{\bar{x}}$	0.018	0.018

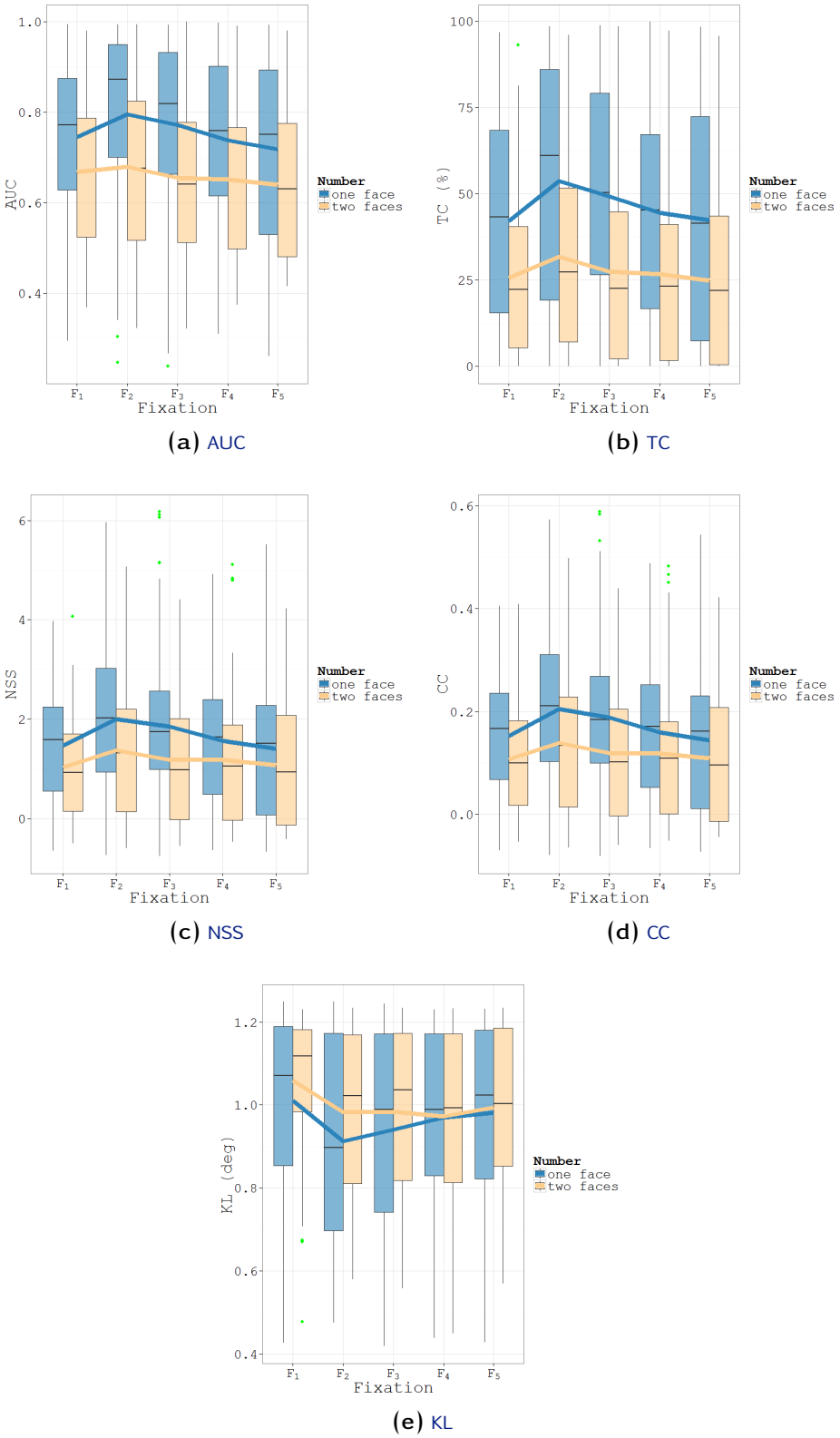
Table 3.3: Comparison criteria for simple M^f and modulated $M^{f'}$ face maps using face data from ground truth evaluated against eye positions density maps. Here, the modulated face maps determine the location of face from the center of the screen.



(a) NSS evolution. (b) CC evolution.

Figure 3.25: Evolution of comparison criteria for simple face maps M^f and modulated face maps $M^{f'}$ using face data from ground truth evaluated against eye positions density maps. The modulated face maps use location of face to the center of the screen rather than the eccentricity from fixation.

into computational models for visual attention.

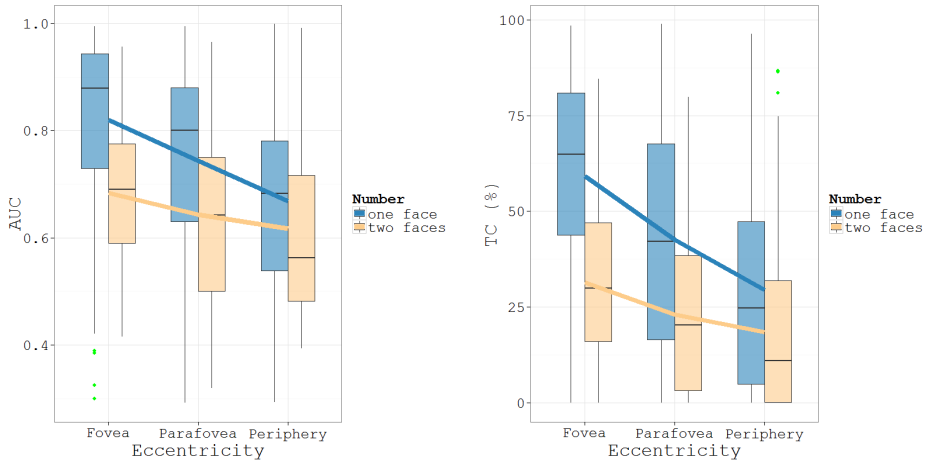


(a) AUC (b) TC

(c) NSS (d) CC

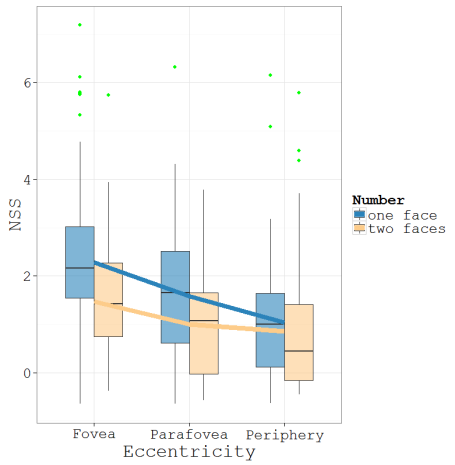
(e) KL

Figure 3.26: Temporal evolution of scores for different evaluation criteria for *'number of faces'*—one and two faces. We took first five fixations $\{F_1, F_2, F_3, F_4, F_5\}$ after current scene onset. The solid lines are the mean scores over time for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

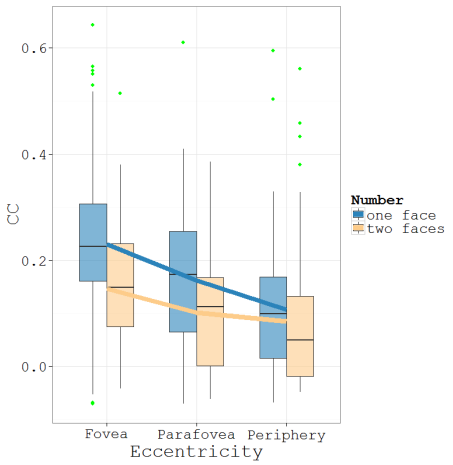


(a) AUC

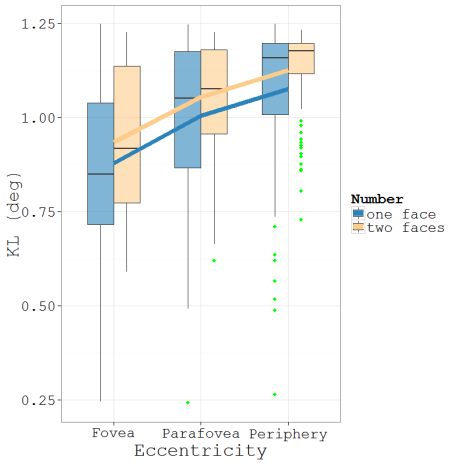
(b) TC



(c) NSS

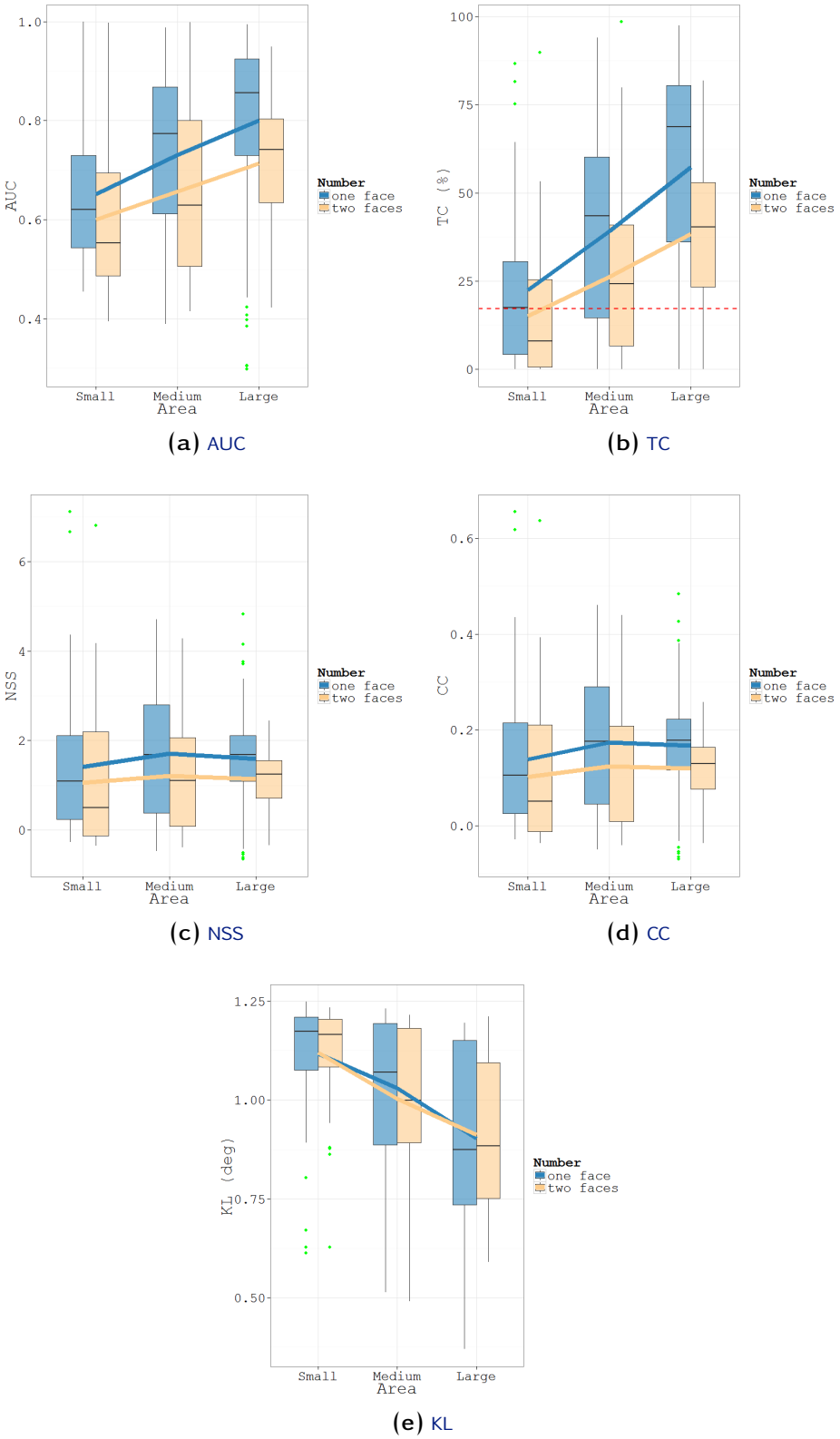


(d) CC



(e) KL

Figure 3.27: Scores for different evaluation criteria as a function of *'eccentricity from fovea'* for one and two faces. Faces were categorized using three regions around fovea (previous fixation): $0° - 2°$ fovea, $2° - 7°$ parafovea and above $7°$ is periphery. In both cases, we took at most five fixations (first five after scene onset). The solid lines are the mean scores for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.



(a) AUC (b) TC

(c) NSS (d) CC

(e) KL

Figure 3.28: Scores for different evaluation criteria as a function of *'face area'* for one and two faces. Faces were categorized into three equally distributed area categories: $0^{(\circ)^2} - 50^{(\circ)^2}$ small, $50^{(\circ)^2} - 75^{(\circ)^2}$ medium and above $75^{(\circ)^2}$ large faces. The area of the face at minimum eccentricity from fovea was considered in the case of two faces. In both cases, we took at most five fixations (first five after scene onset). The solid lines are the mean scores for one and two faces. The box plots show smallest non-outlier observation, lower quartile, median, upper quartile, largest non-outlier observation and outliers.

Table 3.4: Significance analysis to test main and interaction effects of different influencing factors: number of faces, eccentricity of face, area of face and closeness between two faces. Significance codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

			Inter-observer congruency (IOC)				Minimum fixation distance				Comparison criteria (AUC)				Fixation duration			
		d_f	F	p	η_p^2		F	p	η_p^2		F	p	η_p^2		F	p	η_p^2	
	N	189	4.588	0.033	0.024	*	2.05	0.154	0.01		27.05	0	0.14	***	3.306	0.071	0.02	.
One face	E	109	88.87	0.001	0.81	**	88.87	0	0.81	***	29.92	0	0.27	***	6.55	0.012	0.06	*
	A	109	49.9	0.001	0.46	**	49.9	0	0.46	***	54.07	0	0.50	***	0.363	0.548	0.003	
	E*A	107	59.27	0.001	1.66	**	59.27	0	1.66	***	29	0	0.81	***	14.71	0	0.41	.
Two faces	E	78	0.24	0.625	0.003		52.97	0	0.68	***	0.116	0.734	0.001		0.049	0.827	0	
	A	78	0.50	0.48	0.006		38.85	0	0.50	***	9.803	0.002	0.13	**	0.039	0.843	0	
	C	78	14.09	0.001	0.18	**	0.708	0.403	0.009		8.37	0.005	0.11	**	0.008	0.926	0	
	E*A	76	0.521	0.67	0.02		28.12	0	1.11	***	5.014	0.003	0.20	**	0.182	0.908	0.007	
	E*C	76	5.011	0.003	0.2	**	17.3	0	0.68	***	3.63	0.017	0.14	*	3.257	0.026	0.13	*
	A*C	76	5.446	0.002	0.21	**	13.21	0	0.52	***	6.874	0	0.27	***	0.947	0.422	0.04	
	E*A*C	72	2.909	0.01	0.28	*	11.73	0	1.14	***	4.823	0	0.47	***	3.187	0.005	0.31	**

Table 3.5: Paired samples t-tests (2-tailed) to show significant differences for between the two cases of faces (one face and two faces) for following evaluation measures: minimum fixation distance, comparison criteria and fixation duration. Significance codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1.

			Minimum fixation distance			Comparison criteria (AUC)			Fixation duration		
		d_f	t	p		t	p		t	p	
Fixation(#)	F_1	78	0.282	0.779		1.690	0.095	.	-0.904	0.369	
	F_2	77	0.954	0.343		2.982	0.004	**	0.547	0.586	
	F_3	78	1.216	0.228		3.444	0.001	**	-0.51	0.611	
	F_4	74	1.282	0.204		2.207	0.03	*	0.511	0.611	
	F_5	64	2.733	0.008	**	2.571	0.012	*	-0.002	0.999	
Eccentricity	Fovea	59	2.373	0.021	*	3.4086	0.001	**	1.872	0.066	.
	Parafovea	181	1.384	0.168		4.567	0	***	0.084	0.933	
	Periphery	135	0.854	0.394		2.87	0.005	**	-1.472	0.143	
Area	Small	103	2.39	0.019	*	0.6462	0.52		0.862	0.391	
	Medium	122	2.738	0.007	**	3.672	0	***	1.362	0.176	
	Large	150	2.33	0.021	*	5.06	0	***	-2.055	0.041	*

Table 3.6: Paired samples t-tests (2-tailed) to show significant differences for between the two cases of faces (one face and two faces) for inter-observer congruency IOC. Significance codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1.

		Inter-observer congruency (IOC)			
		d_f	t	p	
Fixation(#)	F_1	165	-0.696	0.487	
	F_2	169	-2.592	0.01	*
	F_3	171	-2.167	0.031	*
	F_4	162	-2.328	0.021	*
	F_5	134	-1.07	0.286	
Eccentricity	Fovea	58	-1.319	0.192	
	Parafovea	62	-0.16	0.873	
	Periphery	65	-2.6938	0.009	**
Area	Small	9	-0.810	0.439	
	Medium	114	-2.666	0.009	**
	Large	62	-0.233	0.816	

What information consumes is rather obvious: it consumes the attention of its recipients. Hence a wealth of information creates a poverty of attention, and a need to allocate that attention efficiently among the overabundance of information sources that might consume it.

Herbert Simon

Chapter 4

Visual saliency model with face feature

Numerous visual attention models have been proposed to predict eye movements for static images [KU85; Tso+95; IKN98; HPGGD10; Yan+11], or for dynamic images [LMLCB07; PI08; Mar+09; Mit+10]. Most of them are based on low-level image features (such as color, orientations and spatial frequencies, motion), despite the fact that high-level stimulus properties (e.g., semantic information) also play an important role in visual perception. A recent paper [CFK08] demonstrates that a combined model of high-level object detection and low-level saliency significantly outperformed a low-level saliency model in predicting eye movements. Other studies work on adding a face detection algorithm to increase the saliency at the location of a face [Cer+07; Ma+05]. In fact, [BBK09] finds that visual saliency computed through a classical visual attention model, similar to the one proposed by Itti and Koch [IKN98], does not explain human eye fixations when looking at videos with complex social scenes. Moreover, they conclude that observers often direct their initial gaze toward the eyes and heads of people present in the scenes, and these elements are not emphasized using a classical visual attention model.

Based on all these reported studies, it appears interesting to add *'face features'* into existing visual attention models to improve their efficiency to predict eye movements. This inclusion of faces in models was already done in [Cer+07; Che+03] using static visual stimuli, and in [Ma+05] using dynamic stimuli but for a specific application—video summarization. However, none of the studies incorporate faces in the dynamic visual attention model and compare model predictions to eye movements recorded during free viewing of videos. The aim of the present research was to study the impact of faces on the recorded eye movements of observers looking at videos with various types of content and to examine whether the face feature is biased toward the center of the stimuli, as are the other elementary visual features. Our main contributions are (1) a saliency model that combines low-level feature extraction (static features with orientations and spatial frequencies, dynamic features with object motion amplitude) with a higher-level feature: face, and the comparison to eye movements in videos and (2) the analysis of the impact of the *'center bias'* on the face feature.

In this study, we investigated the influence of faces in dynamic stimuli (videos with various content). We used the Viola-Jones face detector [VJ04] to extract faces in a visual scene along their confidence scores. These scores were put to work to reject bad face detections, and to build a face saliency map using the detected faces. To sum up, the proposed model

introduces a face as an important information feature which is extracted in parallel alongside other classical visual features (static and dynamic features). The improvement of performance of the spatio-temporal model using face features is critical because,

- ❖ In the recorded eye positions, we observed that faces attract gaze [MGP09]. This behavior toward faces was anticipated, as it has already been studied in the context of free-viewing of static images [Cer+07].
- ❖ In the spatio-temporal model, faces were not emphasized enough either in static or dynamic saliency maps. In the static pathway, the textured parts of a face (eyes, nose, mouth) can be salient, but their saliency could be outperformed by salient regions in the background, while for dynamic pathway, faces could be salient, unless they are moving.

4.1 Related work

Face perception is an active field of research, which over the years have interested people from varying fields of research. These areas include visual cognition, neuroscience, and computer vision. Different studies claim that face detection is coded in a specific cortical area of the brain; the fusiform face area (FFA) [KY06; Mec+04; Sum+06]. Electrophysiological studies show that face processing is very fast, and human faces evoke a negative potential at 172ms (N170) [Ben+96].

A recent study [Cer+07] shows that eye movements during free viewing of static natural scenes are largely attracted by faces. Almost 80% of the participants focus on a face within the first two eye fixations. Based on the interest of faces, many visual saliency models are improved by the inclusion of a face channel. The very first attempts are made to improve Itti's model [IKN98] using still images [Cer+07; Ton+10]. The face detector used is Viola-Jones object detector [VJ04]. Other models use mean census transform (MCT) [SS12] and OKAO vision library [SMS12] for face features. Apart from these complex face channel, a simple face channel using skin-color model is proposed to find face candidate regions [WWZ10]. In conclusion, for still images, a comparative study [SCH09] shows improvements in GBVS [HKP07], Itti [IKN98] and GAFFE [Raj+08] visual saliency models, when a face channel is added to them.

Very few studies have used dynamic scenes to evaluate the face channel. The first proposed model [Ho+03] for dynamic scenes uses intensity, color, motion and face features. The proposed face channel uses skin-color model to find face regions in dynamic scenes. Another model [AD07] for dynamic scenes uses several feature maps: contrast-based saliency map, location-based saliency map from the camera motion in the dynamic scenes, and face-based saliency map using detections from an online face detection algorithm.

Some models consider face channel as top-down information [BSL03; SCH08]. Others use the face information alongside other top-down cues. One such saliency model [Kuc11] using contextual and static cues is added with a module for faces, and compared to Itti's model and Hu's model using still images. Another more comprehensive saliency model [TRP07] uses two subsystems, each for bottom-up and top-down pathways respectively. The first subsystem is inspired from Itti's model using static modalities like color, intensity and orientations, while the second uses face and object information. Both systems interact with each other to get saliency. In this model, faces are detected using a statistical skin-color model. All these studies conclude that face channel gives an improvement in predictions.

A model with face channel is used for many applications, for example the detected face regions are used for a conversational agent [Pic+07], video analysis [LN08], image discriminability [Cer+09], retargeting [Dua+11], video surveillance [Gur+10; GC11], and many more.

The organization of the chapter is as follows. In Section 4.2, we present the visual saliency model with three pathways for static, dynamic and face features. We emphasize on the face pathway using Viola-Jones algorithm [VJ04] for face detection in Section 4.2.5. We modulate the resulting saliency maps from all three pathways with center bias in Section 4.3. Section 4.4 describes the evaluation of the model, and summarizes the contributions of adding separate face features to a visual saliency model. We conclude with a discussion in Section 4.6.

4.2 Three-path visual saliency model

The bottom-up algorithm [Mar+09] implemented here is inspired from the human visual system, and is modeled all the way from the retina to visual cortex cells as shown in Figure 4.1. This model can be sub-divided into three distinct pathways: the static pathway, the dynamic pathway, and the face pathway.

4.2.1 Retina-like filters

The retinal model is primarily based on the primate's retina, which imitates the photoreceptor, horizontal, bipolar, and ganglion cells. To begin with, the photoreceptor cells carry out an adaptive compression process on the initial image, modeled as a low-pass Gaussian filter.

$$y = \frac{255 + x_0}{x + x_0} \cdot x$$
$$where\ ,\ x_0 = 0.1 + \frac{410g}{g + 105}$$

The output of photoreceptor cells P is passed on as input to the Horizontal cells, also a function of low-pass Gaussian filter. The response from these cells is twice than the previous retinal low-pass filter.

Down the line are the bipolar cells acting as a high-pass filter, which simply calculates the difference between outputs from Photoreceptor cells P and Horizontal cells H. The bipolar output can be designed to consist of two modes: if *'ON'* then positive part of the difference is kept, otherwise the absolute value when *'OFF'*.

$$Y = ON - OFF$$
$$\text{where,}\quad \begin{matrix} ON & = |P - H| \\ OFF & = |H - P| \end{matrix}$$

The model produces two types of outputs: the parvocellular output that enforces equalization of the visual information by increasing its contrast, consequently, increasing the luminance of low intensity parts in the visual. Next in order is the magnocellular output that responds to higher temporal and lower spatial frequencies.

Analogous to primate's retina, the ganglion cells respond to high contrast and the parvocellular output highlights the borders among the homogeneous regions, thus exposing more detail in the visual.

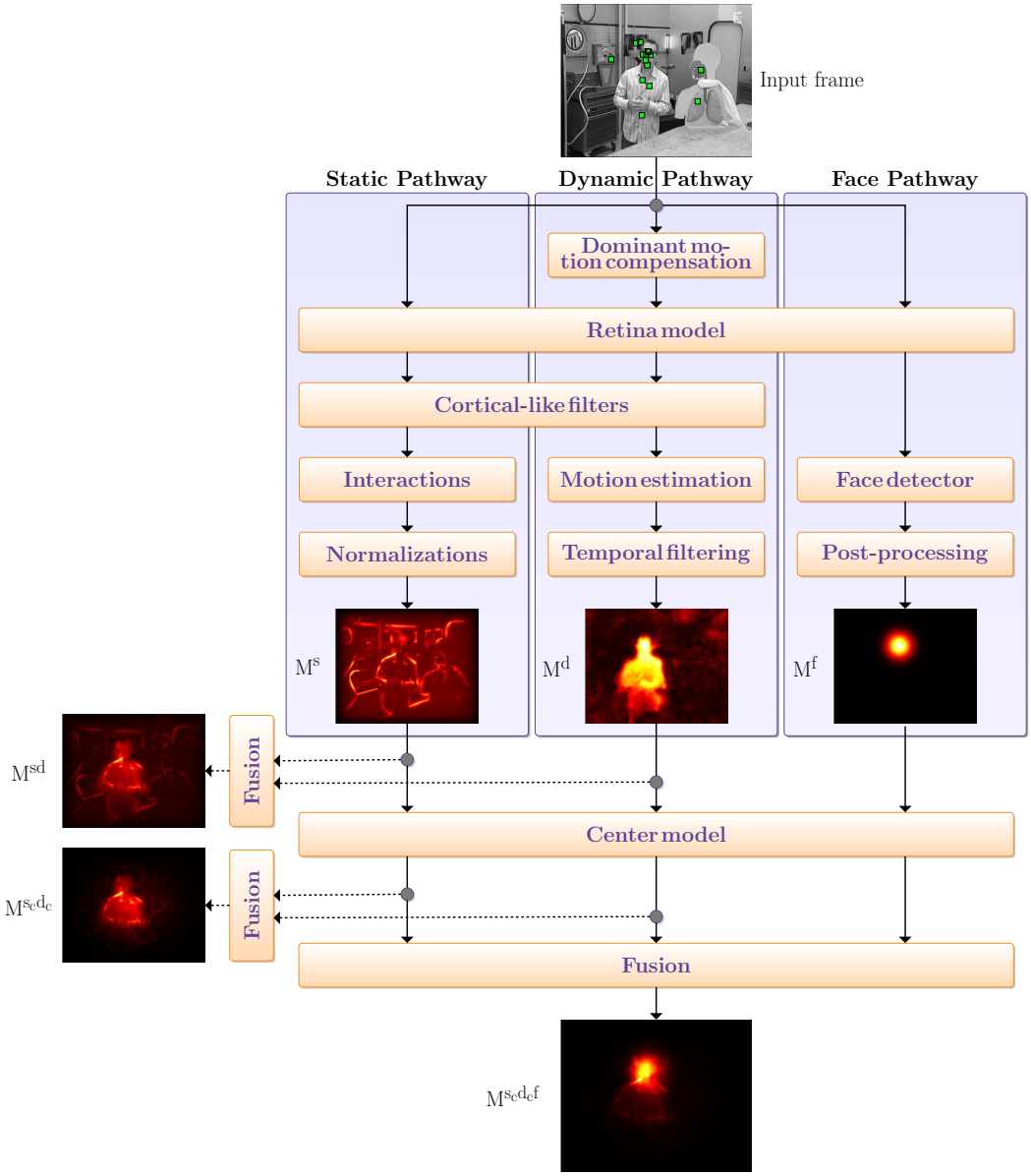


Figure 4.1: Block diagram of the proposed visual saliency model with three saliency maps dedicated to specific features: static, dynamic, and face. All these features are computed in parallel pathways, and resultantly each produces a saliency map—such as M^s, M^d, and M^f. The maps may then be fused together either before or after applying the center model to analyze the influence of the center bias. Here, $M^{s_c d_c f}$ is the final saliency model that combines all the three features with center bias.

4.2.2 Cortical-like filters

The primary visual cortex is a model of simple cell receptive fields that are sensitive to visual signal orientations and spatial frequencies. This can be imitated using a bank of Gabor filters organized in two dimensions, that is closely related to the processes in the primary visual cortex. A Gabor function is defined as:

$$G_{u,v} = exp\left\{-\left(\frac{(u'-f_0)^2}{2\sigma_u^2} + \frac{v'^2}{2\sigma_v^2}\right)\right\}$$
$$\text{where, } u' = u\cos\theta + v\sin\theta$$
$$v' = v\cos\theta - u\sin\theta$$

The retinal output is convolved to the Gabor filters in frequency domain, after applying a mask. The mask is similar to a Hanning function to produce non-uniform illumination approaching zero at the edges. The visual information is processed in different frequencies and orientations in the primary cortex i.e. the model use six orientations and four frequencies to obtain 24 partial maps $M_{u,v}$. These filters demonstrate optimal localization properties and good compromise of resolution between frequency and spatial domains.

4.2.3 Static pathway

The static pathway is based on retinal filtering, which then is followed by a bank of Gabor filters. The main modality used here is intensity.

Interactions

In primate visual system, the response of cell is dependent on its neuronal environment; its lateral connections. Therefore, this activity can be modeled as linear combination of simple cells interacting with its neighbors. This interaction may be inhibitory or excitatory depending on the orientation or the frequency: excitatory when in the same direction, otherwise inhibitory.

$$M_{u,v} = M_{u,v} \cdot \omega$$
$$where,\ \omega = \begin{Bmatrix} 0.0 & -0.5 & 0.0 \\ 0.5 & 1.0 & 0.5 \\ 0.0 & -0.5 & 0.0 \end{Bmatrix}$$

The produced maps are the image's energy in function of the spatial frequency and orientation; after taking into account the interactions among different orientation maps $M_{u,v}$.

Normalizations

The intermediate energy maps from the visual cortical filters and interaction phase are normalized. This model uses a technique [IKN98] proposed by Itti for strengthening the intermediate maps $M_{u,v}$.

- the maps $M_{u,v}$ are normalized to [0 1].
- the maps $M_{u,v}$ are multiplied by $(max(M_{u,v}) - \overline{M_{u,v}})^2$.
- all values lying outside a threshold $\tau = 0.2$ are set to zero.

Summation

Ultimately, a saliency map for the static pathway is extracted for the input visual, simply by summing up all the energy maps. It is significant that the resulting map has salient regions; one's with highest energy, which can be observed in Figure 4.2b by energy located on objects appearing to be salient.

$$M^s = \sum M_{u,v}$$

4.2.4 Dynamic pathway

Humans see stable and moving components in a movie effortlessly. This is true for the case when an object tracked by a camera is seen as moving even if it is stationary on frames. Therefore, we assumed that the human gaze is attracted by motion contrast, which is the motion of a region against its neighbors. The dynamic pathway starts with the estimation and compensation of relative motion using the 2D motion estimation algorithm developed in [OB95]. At the output of this algorithm camera motion is compensated, and then the retina and the Gabor filters allow moving objects to be extracted.

Motion estimation

A differential approach was used for motion estimation that relies on the assumption of luminance constancy. The motion at location (x, y) in a frame t is given by a vector $V(x, y, t)$, which satisfies the optical flow constraint equation:

$$\nabla I(x, y, t) \cdot V(x, y, t) + \frac{\partial I(x, y, t)}{\partial t} = 0$$

For every frame, the optical flow constraint was applied to each Gabor filter output in the same frequency band. This resulted in an over-determined system of equations, which overcomes the aperture problem [BP02]. A motion vector was defined (per pixel) by its modulus and angle—corresponding to the motion amplitude and direction respectively. We only used the modulus of the motion vector to define the saliency of an area, assuming that the motion saliency map of a region is proportional to its speed against the background.

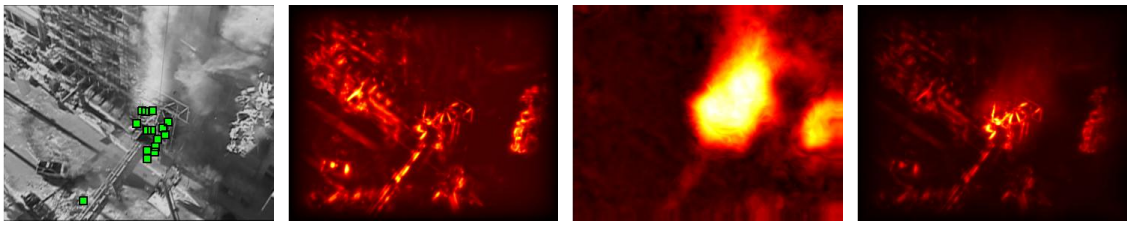
Temporal filtering

If a pixel moved in one frame but not in the previous ones, it is probably the noise resulting from motion estimation. Hence, a temporal median filter was applied to N successive frames—the current one and the five previous frames—to remove this possible noise. Finally, a dynamic saliency map M^d was obtained for each frame (Figure 4.2c).

$$M^d = MED((I_n(i, j))_{0 \leqslant n \leqslant N})$$

4.2.5 Face pathway

The working of the pathway is divided into three steps as follows:



(a) (b) (c) (d)

Figure 4.2: From left to right the input frame with superimposed human eye positions, the static saliency map M^s, the dynamic saliency map M^d, the fusion saliency map M^{sd}.

Pre-processing

The Viola-Jones face detection algorithm uses luminance values to extract facial features, which are prone to environmental factors such as ambient illumination. Different image enhancement methods are used to minimize the contrast of regions with over-exposure or under-exposure. In this step, we used the same retina model to extract the high spatial frequencies of the scene, since in the human visual system FFA takes its input from the parvocellular layer of the lateral geniculate nucleus (LGN) [MG06]. Consequently, the treatment improves the robustness and performance of the detection system in varying illumination conditions.

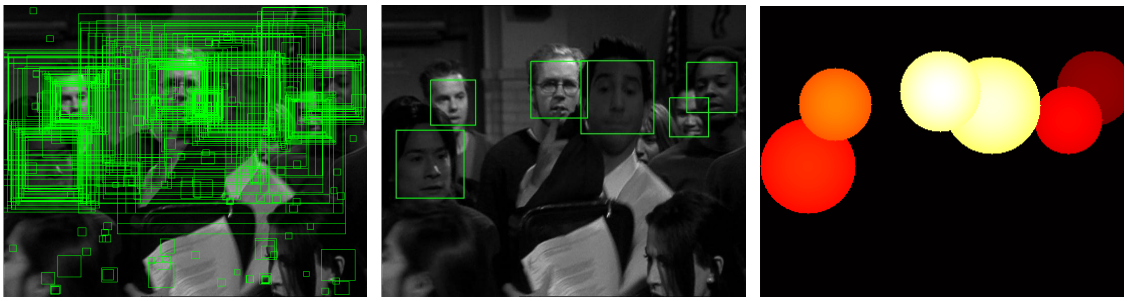
Face detector

The implementation uses the Viola-Jones face detection algorithm from OpenCV library by calling the cvHaarDetectObjects() function. The library also comes with several pre-trained cascade files to detect different types of faces, as summarized in Appendix B. Here, we used the ones for *'frontal'* and *'profile'* faces. The function takes a pretreated gray scale image, and uses a search window to scan across the original image to extract facial features. The search window examines all image locations and classifies them as *'Face'* or *'Not Face'*. This scanning procedure is repeated on several scales to find faces of different sizes by simply resizing the classifier rather than the original image. After the completion of the search process, we obtain multiple neighboring bounding boxes in a positive face region, whereas a single bounding box is often considered as a false detection. The result of the face detector is a set of bounding boxes with a confidence measure and the type of detection—in our case either profile or frontal face. Here, confidence is the measure of existence of an object at a location after all the information has been collected. The estimate is useful to overcome ambiguities by ranking and rejecting several detections. An example of the faces detected is shown in Figure 4.3a.

Post-processing

The face detector was executed twice with two different trained cascades: frontal and profile faces. Hence, the detector returns overlapping face bounding boxes for the same face regions. To judge these face detections, we first computed the overlapping regions among all the face bounding boxes using:

$$A_{i,j} = \sum_{i,j<i} \max \left\{ \frac{h' \cdot w'}{h_i \cdot w_i}, \frac{h' \cdot w'}{h_j \cdot w_j} \right\} > \tau$$



(a) (b) (c)

Figure 4.3: Raw face detections (left), post-processed face detections (middle), face saliency map M^f after post-processing (right)

Here, h and w represent the dimensions of face bounding boxes (both frontal and profile), and h' and w' are the dimensions of their overlapping region. We used a threshold τ of 0.6 or 60% for two face bounding boxes to be considered as overlapping.

This overlapped region A is then used to reject weak detections using their confidence measures c and detected face-type T as follows:

$$Face = \begin{cases} \max(c_i, c_j), & \text{if } T_i = T_j; \\ \max(c_i, c_j), & \text{if } T_i \neq T_j \ \& \ A_{i,j} > 0.8; \\ \min(h_i \cdot w_i, h_j \cdot w_j), & \text{otherwise.} \end{cases}$$

An example of post-processed faces is shown in Figure 4.3b.

Face saliency map

The face saliency map M^f for the face pathway is generated by marking each detected faces bounding box by a 2D Gaussian. The dimensions of the face bounding box determine the distances from its origin in horizontal and vertical axis, while the confidence score is its width or standard deviation. The resulting face saliency map is shown in Figure 4.3c.

The 2D Gaussian on a detected face is determined using the equation.

$$f(x, y) = A \cdot e^{-\left(\frac{x-x_0}{2\sigma_x^2} + \frac{y-y_0}{2\sigma_y^2}\right)}$$

where amplitude A is the confidence of the face, (x_0, y_0) is the center of the Gaussian window and corresponds to the origin of the face bounding box (O_x, O_y), and (σ_x, σ_y) is the standard deviation of the Gaussian determined by the dimensions of the face bounding box h and w.

Three-pathway fusion

The fusion method modulates the static, dynamic and face saliency maps using maximum, skewness and confidence c respectively, and fuses them together using:

$$M^{sdf} = \alpha \cdot M^s + \beta \cdot M^d + \gamma \cdot M^f + \alpha\beta \cdot M^s \cdot M^d + \beta\gamma \cdot M^d \cdot M^f + \alpha\gamma \cdot M^s \cdot M^f$$

$$where, \begin{cases} \alpha = max(M^s) \\ \beta = skewness(M^d) \\ \gamma = mean(c) \end{cases}$$

An example of the fusion is given in Figure 4.4. As mentioned in [Mar+09], the maximum and skewness are appropriate weightings for static and dynamic saliency maps respectively. Similarly, the weighting suitable for face saliency map is its confidence. The higher is the confidence, the greater is the probability of the presence of a face in an image, and hence, the higher is the weight assigned to the map during fusion. However, in Appendix C, we conclude that every fusion method has their separate advantages, and the choice of the method is dependent on the video content, or the application.

The fusion method proposed here uses modulated confidence based on the method presented in Section 3.8. Since the model predicts salient regions rather than eye fixations, we use the distance of the location of face on the screen to the center of screen (G_x, G_y) for the modulation. In the previously presented method, eccentricity of face from fixation was used.

We also assume that common salient regions in different saliency maps obtain the most representation in the final map. This is done by reinforcing such regions by a multiplicative fusion of different feature maps.

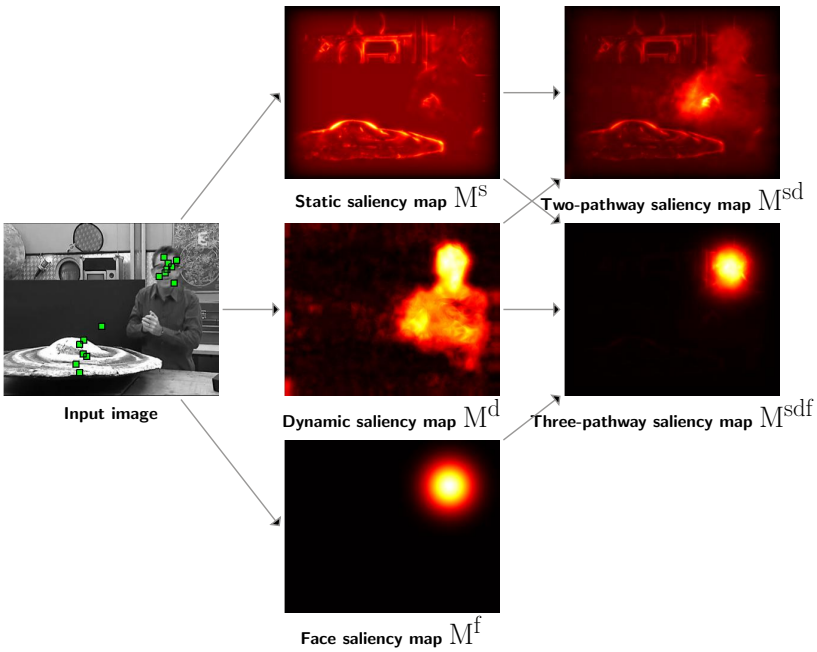


Figure 4.4: Block diagram illustrates the fusion of two and three pathway visual saliency models for video database. The two-pathway saliency map M^{sd} is the result of the fusion of saliency maps M^s and M^d, whereas the three-pathway saliency map M^{sdf} also takes into account the face saliency map M^f alongside the other two saliency maps.

4.3 Model with center bias

The central fixation bias in visual scene viewing is selecting an optimal viewing position independent of the image features. The center might not provide an information-processing advantage, but it is an optimal position to explore the visual scene. A number of factors contribute to this effect [Tse+09; Dor+10; ZK11]: high-level strategic advantage, drop in visual sensitivity in periphery, and motor bias. Moreover, people tend to direct their first saccades in the visual scene toward subjects of interest or salient locations closer to the center, as the initial saccades are a localizing response, and afterward are the explorations of the objects [Tat07; RVC07]. In this study, we examine the influence of center bias on each of the three visual feature maps: static, dynamic, and face. The observations were used to propose a center model.

The center model

The center model is a significant predictor of eye position in arbitrary natural scenes, due to the preference of placement of focal and foreground objects in the center of the screen. This model alone outperforms models without a central bias [LMLCB07; Jud+09; Zha+08]. Its introduction enhances the correlation between eye positions and computational model output, but it might not be useful for applications of visual attention because it is not specific to visual features. The centered model is considered as a saliency model applying the same *'Central'* saliency map M^c to all the frames. In our case, this map corresponds to a 2D Gaussian with sigma 10° and dimensions equal to that of the original video frame. It is applied multiplicatively to the spatial information as illustrated in Figure 4.5.

$$M^{m_c} = M^m \cdot M^c$$

Here, M^m is for the feature maps from different pathways of the visual saliency model.

$$M^c = e^{-\frac{1}{2}(\alpha \frac{n}{N/2})^2}$$

where $-N/2 \leq n \leq N/2$ is the size of the Gaussian window calculated using $\alpha = 1/2.5$.

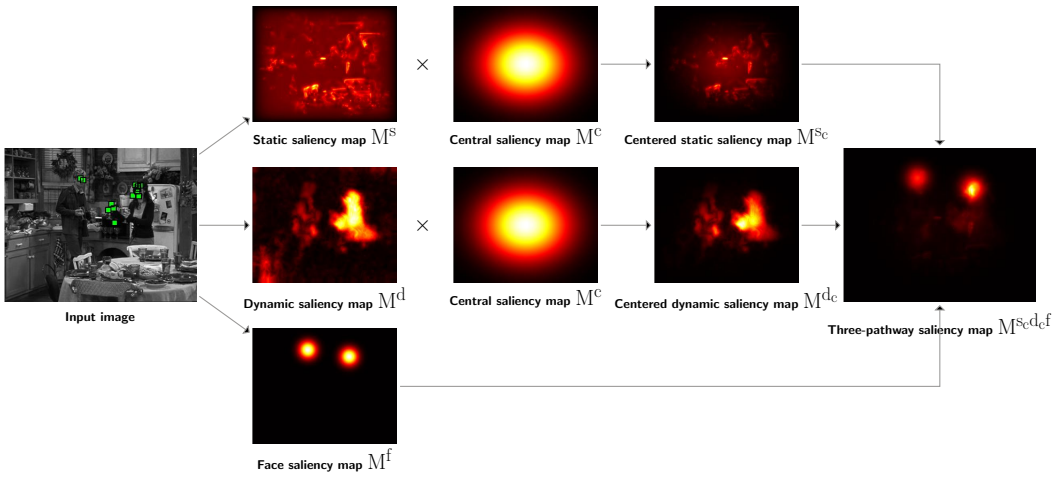
Three-pathway fusion with center bias

The fusion model fuses the centered static saliency map M^{s_c}, the centered dynamic saliency map M^{d_c} and the face saliency map M^f using maximum, skewness and confidence scores respectively.

$$\begin{aligned} M^{s_c d_c f} =& \alpha \cdot M^{s_c} + \beta \cdot M^{d_c} + \gamma \cdot M^f \\ &+ \alpha\beta \cdot M^{s_c} \cdot M^{d_c} + \beta\gamma \cdot M^{d_c} \cdot M^f + \alpha\gamma \cdot M^{s_c} \cdot M^f \end{aligned}$$

The fusion weights α, β and γ are computed for centered static saliency map M^{s_c}, centered dynamic saliency map M^{d_c} and face saliency map M^f respectively. Here, the saliency maps from the face pathway are not modulated with center bias. The reason is the attractiveness of face stimuli regardless of their location around the center of a visual scene. Also, the occasional presence of faces in the video sequences suggests that such weighting did not significantly improve the results of evaluation as shown in Table 4.3.

Figure 4.5: Block diagram illustrates the fusion of saliency maps from the three pathways of the model. The center bias is applied to saliency maps M^s and M^d to obtain centered saliency maps M^{s_c} and M^{d_c}. These two resulting maps are fused to the face saliency map M^f to obtain final saliency map $M^{s_c d_c f}$.



4.4 Results: Evaluation of face pathway

4.4.1 Performance of the face detector

We took the video database used in our experimental design to record eye positions of participants detailed in Appendix A, and used it to evaluate the performance of the face detector. The detection was carried out on all 14,155 input video frames. The resulting faces detected were hand-labeled as either a true or a false face detection (Table 4.1). An example of the faces detected is shown in Figure 4.3a.

Video database	Total detections	True positive	False positive	Percentage of true positives
SM-I	7,696	5,424	2,272	70%
GS-II	10,074	6,954	3,120	69%

Table 4.1: Results of the face detector for our video database.

4.4.2 Interest of separate face pathway

It is important to note that if we do not use a face pathway, fusion of the static and dynamic pathways is not sufficient to make faces salient. As already found in the literature, a classical visual attention model (only static and dynamic features) cannot explain gaze on face locations because they are not always emphasized in such a model. To investigate this point, we performed a simple analysis using the static saliency map and true-positive face detections obtained by comparing the detected faces to the hand-labeled ground truth. We started by calculating the mean static saliency values at face locations for all n frames with at least one face; $\{x_1, x_2, ..., x_n\}$. The mean $\bar{x}$ was used as a threshold to split the frames

into two subgroups: face salient, and face non-salient frames. Subsequently, we used the two metrics to compute the scores for these subgroups of frames. The resulting scores in Table 4.2 show that faces have similar scores in the conditions of low or high static saliency. The comparison criteria are high for salient faces as expected because faces are attractive. Likewise, non-salient faces also have high scores for both metrics. We can imply based on this observation that static saliency maps do not model the saliency of faces, but are instead only activated by object contrasts. Hence, it is interesting to include a separate face pathway to improve the model's eye movement predictions.

		Face salient	Face non-salient	
Samples(#)		1573	2077	
AUC	$\bar{x}$	0.65	0.64	$(F_{(1,221)} = 0.05, p = 0.82)$
	$SE_{\bar{x}}$	0.001	0.001	
TC (%)	$\bar{x}$	32	32	$(F_{(1,221)} = 0.01, p = 0.93)$
	$SE_{\bar{x}}$	0.164	0.161	
NSS	$\bar{x}$	2.08	1.64	$(F_{(1,221)} = 3.21, p = 0.07)$
	$SE_{\bar{x}}$	0.010	0.009	

Table 4.2: *AUC*,*TC* and *NSS* results for the frames with at least one face in a condition of high or low static saliency. Only positive faces from *SM-I* video database are used.

4.4.3 Face pathway with center bias

One wonders which is the most important bias while watching all types of videos; the center bias that tends to make a participant to gaze more at the center, or the particularity of faces which makes participants to look at a face and recognize it more rapidly than other objects. In the former case, Dorr et al. [Dor+10] discussed the impact of the center bias on different types of videos, such as professionally made *'Hollywood movies,'* and amateur made *'natural movies.'* They show an increased impact of center bias in *'Hollywood movies'*—the kind of videos that were used to generate our clips. In the latter case, it is shown in [Her+10] that faces are easier to detect than other objects, and the detection facilitation of such stimuli is higher even if it is presented at the periphery. They concluded that the spatial window for face detection is wider than for other objects. In our experiment, we observed similar results when considering the scores for the face saliency map M^f (NSS = 1.66, AUC = 0.70, TC = 47%) or the one weighted by center bias M^{f_c} (NSS = 1.71, AUC = 0.70, TC = 47%) from Tables 4.4 and 4.5 respectively. This result could either be explained by a smaller impact of center bias for faces than other salient objects or by the fact that there were few faces present in the frames to make the center bias significant on face saliency maps.

To investigate further the probable cause of faces attracting human gaze independent of their location, we calculated scores for the maps M^c and M^{f_c}, as presented in Table 4.3. Here, the frames considered contained at least one face, yet the impact of central weighting is still marginal. Furthermore, there was no impact of the center model on the scores (*NSS*, *AUC* and *TC*) for M^g and M^{g_c}, where we considered only the true-positive detections obtained by comparing the detected faces to the hand-labeled faces (ground truth). Therefore, our results agree with the finding of [Her+10] that the presence of faces attracts the gaze of a participant more than their tendency to fixate on the center. This is not the case for the two

other features; in fact, adding the center bias on the static and the dynamic saliency maps considerably improved the scores.

Video database	Criterion		M^f	M^{fc}	M^g	M^{gc}
SM-I	AUC	$\bar{x}$	0.66	0.66	0.70	0.70
		$SE_{\bar{x}}$	0.006	0.006	0.008	0.008
	TC (%)	$\bar{x}$	39	39	47	47
		$SE_{\bar{x}}$	1.308	1.284	1.654	1.628
	NSS	$\bar{x}$	1.35	1.41	1.76	1.81
		$SE_{\bar{x}}$	0.053	0.051	0.070	0.066
	CC	$\bar{x}$	0.27	0.28	0.35	0.36
		$SE_{\bar{x}}$	0.009	0.008	0.013	0.011
	KL (deg)	$\bar{x}$	0.77	0.75	0.68	0.67
		$SE_{\bar{x}}$	0.010	0.010	0.013	0.013
GS-II	AUC	$\bar{x}$	0.61	0.61	0.65	0.65
		$SE_{\bar{x}}$	0.005	0.005	0.007	0.007
	TC (%)	$\bar{x}$	26	26	33	33
		$SE_{\bar{x}}$	1.094	1.094	1.448	1.448
	NSS	$\bar{x}$	1.33	1.24	1.85	1.72
		$SE_{\bar{x}}$	0.074	0.069	0.103	0.097
	CC	$\bar{x}$	0.29	0.27	0.40	0.37
		$SE_{\bar{x}}$	0.013	0.012	0.019	0.017
	KL (deg)	$\bar{x}$	0.79	0.79	0.68	0.68
		$SE_{\bar{x}}$	0.014	0.014	0.019	0.019

Table 4.3: *AUC*, *TC*, *NSS*, *CC* and *KL* results for face saliency maps (M^f and M^{fc}) with or without the center model. Similarly, M^g and M^{gc} are face saliency maps comprising of only the true-positive face detections. We only considered video frames with at least one face detected.

4.5 Results: Evaluation of the model

4.5.1 Comparison criteria scores

Table 4.4 presents the results for different saliency maps. Here, we calculated the sample mean $\bar{x}$ for first 50 frames for all 305 video snippets and then took the standard deviation of this sample mean denoted as $SE_{\bar{x}}$. As shown in [Mar+09], the dynamic saliency map M^d performs better than the static one M^s for both criteria. This is due to the importance of motion in guiding attention [Mit+10; WH04]. The lower results for the face saliency maps M^f are explained by the fact that faces are present only in a small percentage of the video database (35%), thus, and for the rest of the frames, the prediction score is zero. Moreover, the fusion integrating the face pathway M^{sdf} outperforms the fusion combining static and dynamic pathway M^{sd}. M^{sdf} takes into account the face information when there is a face, otherwise it is similar to M^{sd} when no face is detected. This additional information considerably improves the results because face is a powerful gaze-attractor.

In Table 4.4, the saliency maps for all pathways have lower prediction scores than a simple Gaussian at the center of the frame—central saliency map M^c. This demonstrates that center bias has an impact on eye positions during free viewing, and its appropriate integration will improve the model's efficiency in predicting eye movements.

Video database	Criterion		M^c	M^s	M^d	M^f	M^{sd}	M^{sdf}
SM-I	AUC	$\bar{x}$	0.81	0.65	0.67	0.70	0.65	0.66
		$SE_{\bar{x}}$	0.004	0.003	0.002	0.004	0.002	0.003
	TC (%)	$\bar{x}$	65	50	47	52	55	71
		$SE_{\bar{x}}$	0.924	0.808	0.334	0.771	0.474	0.668
	NSS	$\bar{x}$	1.29	0.74	0.93	1.08	1.00	1.38
		$SE_{\bar{x}}$	0.024	0.015	0.015	0.031	0.016	0.022
	CC	$\bar{x}$	0.30	0.16	0.18	0.20	0.20	0.29
		$SE_{\bar{x}}$	0.005	0.003	0.002	0.004	0.002	0.004
	KL	$\bar{x}$	0.50	0.53	0.55	0.54	0.51	0.50
		$SE_{\bar{x}}$	0.001	0.001	0.001	0.002	0.001	0.001
GS-II	AUC	$\bar{x}$	0.71	0.62	0.59	0.57	0.63	0.65
		$SE_{\bar{x}}$	0.006	0.002	0.002	0.003	0.002	0.005
	TC (%)	$\bar{x}$	47	44	35	32	45	50
		$SE_{\bar{x}}$	0.942	0.785	0.395	0.727	0.292	0.915
	NSS	$\bar{x}$	0.82	0.57	0.45	0.38	0.72	0.91
		$SE_{\bar{x}}$	0.027	0.014	0.010	0.025	0.011	0.023
	CC	$\bar{x}$	0.23	0.15	0.11	0.08	0.18	0.24
		$SE_{\bar{x}}$	0.006	0.003	0.003	0.005	0.002	0.006
	KL	$\bar{x}$	0.50	0.52	0.53	0.53	0.51	0.49
		$SE_{\bar{x}}$	0.001	0.001	0.001	0.001	0.001	0.001

Table 4.4: *AUC*, *TC*, *NSS*, *CC* and *KL* results for the different pathways of the model without the center bias, except M^c that represents the center model. The sample mean $\bar{x}$ and its standard error $SE_{\bar{x}}$ were averaged over the 305 video snippets and 80 video snippets for *GS-II* video database. Blank maps are not considered in mean scores for M^f.

4.5.2 Influence of center bias

Figure 4.6 is an illustration of all human eye positions from the experiment superimposed as a 2D-image. We clearly observed that there is an apparent center bias effect in play. This effect could be the result of the experimental setup where the first eye position for a clip starts from the central marker. Also, it could be a significant contribution due to the video content presented such as *'Hollywood movies'* [Dor+10]. Consequently, this motivates us to incorporate a center model to enhance the relevance among experimental eye positions and visual saliency maps from the proposed model. In fact, practical applications potentially using the visual attention model might prefer to use the entire information, and hence not require this modulation.

Human data is highly center-biased as shown in Figure 4.6. Hence, adding a larger border will increase the overall performance of the model. The result is consistent with the fact that saliency falls off with the distance from the center. As a result, we find that the simple

Gaussian technique outperformed the results of the model without center bias. Consequently, the fusion step then considered the nature of each map and integrated a center bias when appropriate to reinforce the salient regions. The resulting master saliency map performed better than each pathway predicting independently and standalone Gaussian map from the central model.

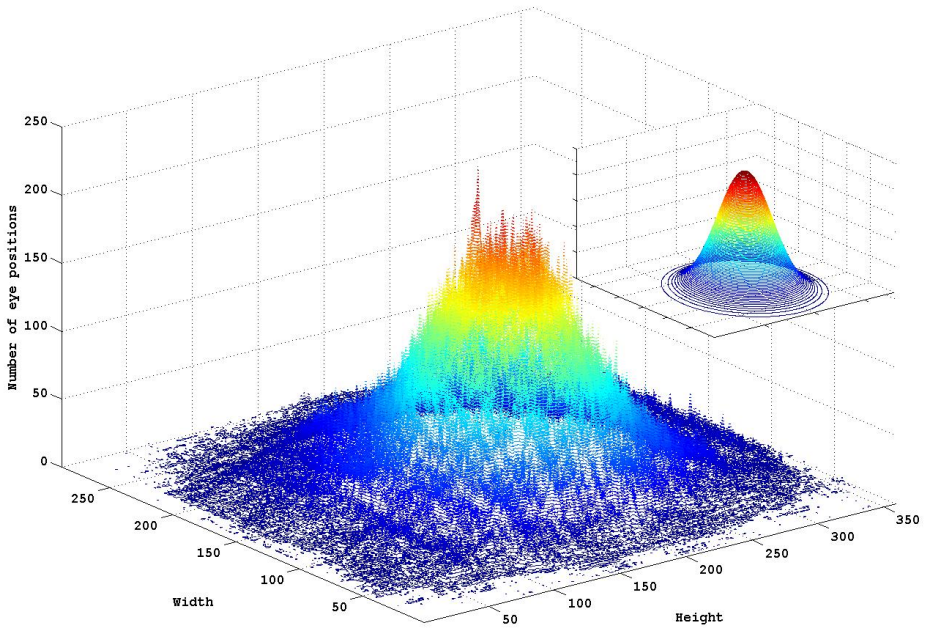


Figure 4.6: 2D contour map presents the distribution of participants eye positions for video database, and the distribution after Gaussian fitting is shown in subplot.

4.5.3 Model with center bias

Table 4.5 shows that center bias modulation of the saliency maps' results in higher scores for all maps such that M^{s_c} outperforms M^s ($F_{(1,609)} = 178.59$, $p < 0.001$)[1] and gives similar results to M^c. The results between M^c and M^{s_c} are very close, even if M^c still outperforms M^{s_c} with respect to the *NSS* criteria ($F_{(1,609)} = 4.62$, $p < 0.05$). M^{d_c} outperforms both M^d ($F_{(1,609)} = 87.36$, $p < 0.001$) and M^c ($F_{(1,609)} = 18.98$, $p < 0.001$). More than looking at the center of the frame a participant gazes at what is salient near the center of the frame, since the static and dynamic saliency maps both provide complementary information that is needed to predict the participant eye positions. In addition to the integration of center bias before its fusion into the saliency map $M^{s_c d_c}$ outperforms the simple fusion saliency map M^{sd} ($F_{(1,609)} = 65.25$, $p < 0.001$) and central saliency map M^c ($F_{(1,609)} = 4.93$, $p < 0.05$). Similarly, the fusion of the three pathways with center model $M^{s_c d_c f}$ gives the best results, outperforming the central saliency map M^c ($F_{(1,609)} = 45.72$, $p < 0.001$), centered two-pathway saliency maps $M^{s_c d_c}$ ($F_{(1,609)} = 19.40$, $p < 0.001$), and three-pathway saliency maps M^{sdf} ($F_{(1,609)} = 34.70$, $p < 0.001$). Therefore, both center bias and faces are important to consider to obtain a good predictor of eye positions of participants.

[1] For clarity, only statistics using *NSS* criteria are presented since all metrics (*AUC*, *TC*, *NSS*, *CC* and *KL*) generally produce the same conclusion. We took the sample mean for first 50 frames from each video snippet, and then applied the significance tests.

Video database	Criterion		M^c	M^{s_c}	M^{d_c}	M^{f_c}	$M^{s_c d_c}$	$M^{s_c d_c f}$
SM-I	AUC	$\bar{x}$	0.81	0.70	0.75	0.66	0.67	0.70
		$SE_{\bar{x}}$	0.004	0.003	0.002	0.006	0.004	0.004
	TC (%)	$\bar{x}$	65	67	67	39	57	72
		$SE_{\bar{x}}$	0.924	0.810	0.449	1.279	0.655	0.689
	NSS	$\bar{x}$	1.29	1.22	1.53	1.41	1.18	1.61
		$SE_{\bar{x}}$	0.024	0.022	0.016	0.051	0.029	0.031
	CC	$\bar{x}$	0.30	0.27	0.32	0.28	0.23	0.33
		$SE_{\bar{x}}$	0.005	0.004	0.002	0.008	0.005	0.005
	KL	$\bar{x}$	0.50	0.50	0.51	0.75	0.47	0.47
		$SE_{\bar{x}}$	0.001	0.001	0.001	0.010	0.001	0.002
GS-II	AUC	$\bar{x}$	0.71	0.64	0.71	0.61	0.64	0.67
		$SE_{\bar{x}}$	0.006	0.005	0.004	0.005	0.003	0.005
	TC (%)	$\bar{x}$	47	49	51	26	47	54
		$SE_{\bar{x}}$	0.942	0.843	0.819	1.109	0.390	0.891
	NSS	$\bar{x}$	0.82	0.81	0.94	1.24	0.99	1.19
		$SE_{\bar{x}}$	0.027	0.022	0.021	0.070	0.029	0.031
	CC	$\bar{x}$	0.23	0.22	0.25	0.27	0.24	0.30
		$SE_{\bar{x}}$	0.006	0.005	0.005	0.005	0.006	0.007
	KL	$\bar{x}$	0.50	0.50	0.50	0.79	0.49	0.47
		$SE_{\bar{x}}$	0.001	0.001	0.001	0.014	0.001	0.002

Table 4.5: *AUC*, *TC*, *NSS*, *CC* and *KL* results for the different pathways of the model with the center bias, except M^c that represents the center model. The sample mean $\bar{x}$ and its standard error $SE_{\bar{x}}$ were averaged over the 305 video snippets and 80 video snippets for *GS-II* video database. Blank maps are not considered in mean scores for M^{f_c}.

4.5.4 Choice of coefficients for two pathways

The static and dynamic saliency maps were not normalized, and the raw values were used to evaluate the pertinence of the static map. To quantify how the maximum and skewness characterize static and dynamic maps respectively, we did some computations and comparisons to the experimental eye data. We divided our frames according to the maximum and the skewness values computed on the static and the dynamic maps. We classified video frames according to their maximal value computed on their static saliency maps. We split our database in two groups: maps with a high maximum value (static saliency map), and the maps with low maximum value. Similarly, we split our database into two groups according to the skewness value computed for the dynamic saliency maps. Note that these classifications were done independently.

For the static saliency maps, the mean would reflect more the quantity of the salient regions, whereas the maximum is more able to express the power of the most salient region of a map. The scores in Table 4.6 show that frames with a high maximum value for static saliency map better explains eye movements than the frames with a low maximum value. This effect decreases when using the skewness of the static saliency map. In consequence, we selected the better metric, the maximum, as a weighting coefficient for the static saliency map during fusion. In case of the dynamic saliency maps, we compute the distribution of pixels using

skewness. We observe that the maps have higher skewness when there is a small object in motion, and we wanted to enhance such maps as they strongly predict eye positions. Hence, we chose the skewness to weight the dynamic saliency maps for the fusion, because high skewness reflects a better predictability of eye positions (both *NSS* and *TC* metric scores in Table 4.6 are higher for frames with the higher skewness).

			High maximum	Low maximum	High skewness	Low skewness
M_s	NSS	$\bar{x}$	0.75	0.74	0.73	0.75
		$SE_{\bar{x}}$	0.014	0.006	0.010	0.006
	AUC	$\bar{x}$	0.60	0.66	0.60	0.68
		$SE_{\bar{x}}$	0.002	0.001	0.001	0.001
	TC (%)	$\bar{x}$	54	49	52	48
		$SE_{\bar{x}}$	0.431	0.228	0.330	0.253
M_d	NSS	$\bar{x}$	0.73	1.12	1.47	0.57
		$SE_{\bar{x}}$	0.000	0.000	0.000	0.000
	AUC	$\bar{x}$	0.66	0.68	0.72	0.64
		$SE_{\bar{x}}$	0.000	0.000	0.000	0.000
	TC (%)	$\bar{x}$	42	51	61	37
		$SE_{\bar{x}}$	0.004	0.004	0.005	0.003

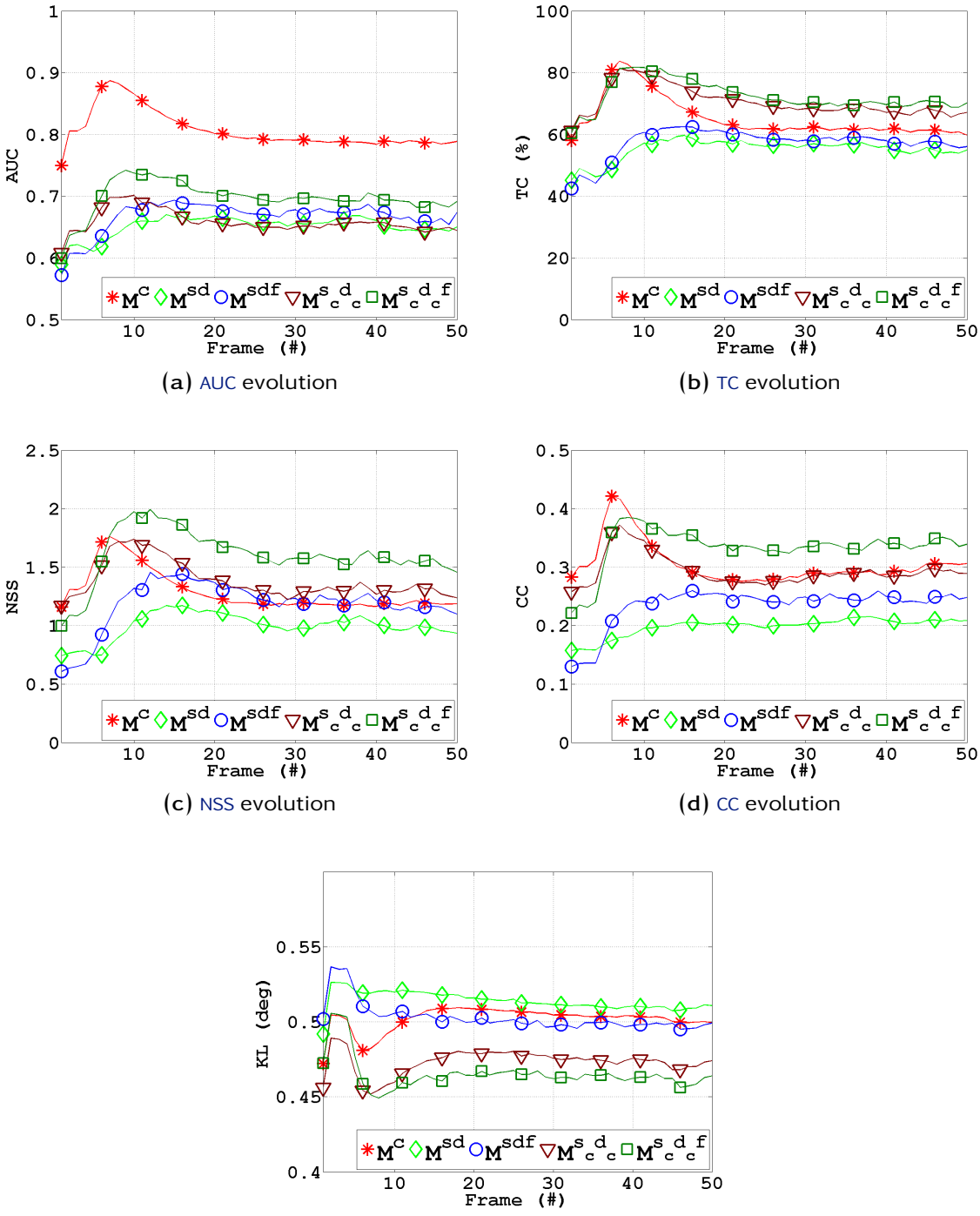
Table 4.6: *NSS*, *AUC* and *TC* results for the frames classified according to their maximal and their skewness values for both static and dynamic saliency maps.

4.5.5 Temporal evolution

The time course of the influence of center bias has also been investigated [Tat07; Tse+09]. To analyze the evolution of the center bias effect on videos, different saliency map scores were plotted along frame position. For each frame position inside a snippet (independently of the snippet position in the clip), all the scores corresponding to that frame position were averaged. We took only the first 50 frames even if some snippets were longer.

Figures 4.7 and 4.8 illustrates that all saliency maps were more predictive at the beginning of each snippet from 5^{th} to 13^{th} frame. Afterward, the prediction scores decreased, which is consistent with the fact that the proposed model is bottom-up. The peaks corresponding to center biased saliency maps $M^{s_c d_c}$ and $M^{s_c d_c f}$ were sharper compared to saliency maps without center bias M^{sd} and M^{sdf}. The biased saliency maps reached their maximum quickly around the 8^{th} frame, and clearly outperformed maps without center bias. However, they decreased more rapidly showing that center bias is particularly predominant at the scene onset, as also mentioned in [Tse+09]. Therefore, the influence of center bias decreases along time, letting other features to take over irrespective of their position in the visual scene.

The temporal evolution of metrics also showed that the introduction of face pathway is an improvement. In Figures 4.7 and 4.8, we find that three-pathway saliency map M^{sdf} did produce better metric scores compared to two-pathway saliency map M^{sd}. Moreover, M^{sdf} performed comparatively well against the two-pathway saliency map with center bias $M^{s_c d_c}$. This result indicated that face feature as a separate pathway certainly did increase the



(a) AUC evolution

(b) TC evolution

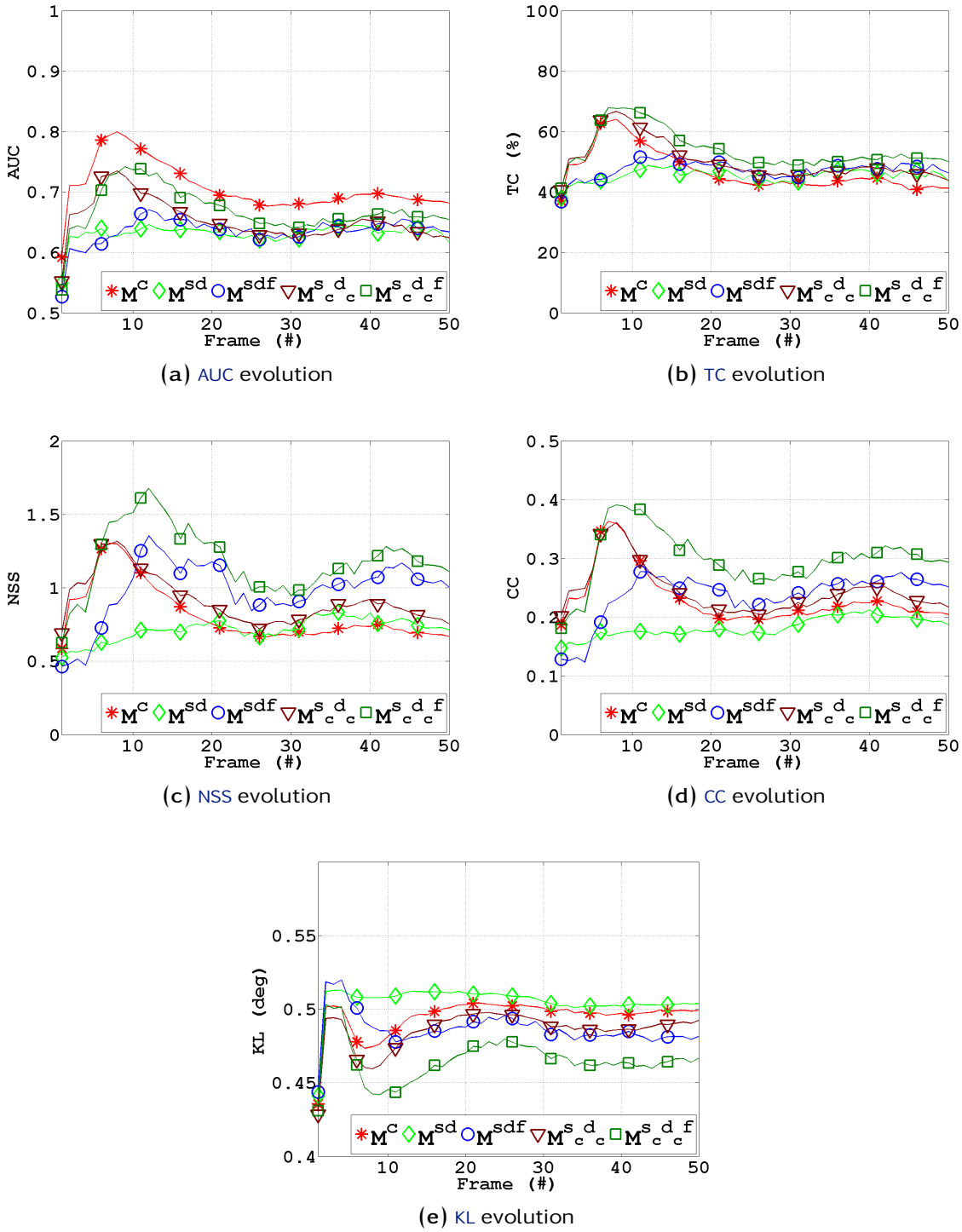
(c) NSS evolution

(d) CC evolution

(e) KL evolution

Figure 4.7: Evolution of metrics (AUC, TC, NSS, CC and KL) for different pathways with or without the center bias for *SM-I* video database.

predictability power of the model. The saliency map was reinforced further by center bias, and the resulting saliency map $M^{s_c d_c f}$ delivered the best scores.



(a) AUC evolution

(b) TC evolution

(c) NSS evolution

(d) CC evolution

(e) KL evolution

Figure 4.8: Evolution of metrics (AUC, TC, NSS, CC and KL) for different pathways with or without the center bias for *GS-II* video database.

4.6 Discussion and conclusion

This study presents a new bottom-up saliency model that breaks down the visual signal using three processing pathways based on different types of visual features: static, dynamic,

and face. The static and dynamic pathways are inspired by the biology of the first steps of the human visual system: a retina-like filter and a cortical-like bank of filters. The static pathway extracts the texture information based on luminance. The dynamic pathway extracts information about objects' motion against background. The face pathway extracts information about the presence of faces in the frames. This model also integrates the center bias as a suitable modulation on the different saliency maps.

An eye movement experiment was used to record the gaze of participants viewing various videos freely. This experiment was used to evaluate and also to improve the saliency model. Each pathway is effective for predicting eye movements. The face pathway is particularly effective for predicting eye movements on frames containing faces, highlighting the importance of integrating face feature in a bottom-up saliency model. The eye movement experiment enables us to study which visual features attract a participant's gaze, and how to integrate them into the saliency model, and more particularly, to the fusion step of the three pathways. The fusion of the three types of maps into a single master saliency map is optimized by weighting the saliency maps produced by the three pathways using specific coefficients. The coefficients correspond to particular statistics extracted from the different types of saliency maps (maximum, skewness, and confidence). These weights are then used to strengthen the most relevant feature maps.

The study concentrates on the importance of faces and center bias for the improvement of a visual saliency model. In future work, we hope to analyze the evolution of performance of the proposed model for longer videos, when bottom-up processes are no longer predominant and top-down processes might play an important role on eye movements. Thus, the bottom-up visual saliency model can be integrated with top-down weights to modulate the saliency maps as a function of the goal. The resulting saliency maps from combined stimulus-driven and goal-driven model can give better prediction of eye movements.

For over a decade prophets have voiced the contention that the organization of a single computer has reached its limits and that truly significant advances can be made only by interconnection of a multiplicity of computers.

Gene Amdahl

Chapter 5

GPU implementation of the model

Often visual saliency models incorporate a number of complex tasks that make a real-time solution quite difficult to achieve. The objective can be achieved only by the simplification of the overall model as done by Itti [Itt05] and Nabil [OH03]. This makes impossible the inclusion of other features into the existing model. Over the years, GPUs have evolved from fixed function architectures into completely programmable shader architectures. All together with a mature programming model like CUDA [Cud] makes the GPU platform a preferable choice for acquiring high performance gains. Generally, vision algorithms are a sequence of filters that are relatively easier to implement on GPU's massively data-parallel architecture. Also, the graphics device is cheaper, accessible to everyone, and simple to program than its counterparts.

In this chapter, we propose an adaptation of the visual spatio-temporal saliency model presented above onto GPU. This model mimics human visual perception from retina to visual cortex using both static and dynamic information to calculate the final saliency map and hence compute-intensive. After this transformation, we apply several optimizations to leverage the raw computational power of graphics hardware. Subsequently, proposing a real-time solution on multi-GPU, and demonstrating the performance gains. In the end, we also evaluate the effects of lower precision on the resulting saliency map of our model.

5.1 Related work

In recent years, a number of researchers have shown their interest to exploit the power of commodity hardware including multiple processor cores for parallel computing for different algorithms. The different architectures offer different advantages over one another depending on two main factors: cost and accessibility. First, multi-core CPUs and stream processors such as graphics processing units (GPUs) [Cud; Amd] and Cell processors [Cel] are powerful, efficient and economical. Second, the availability of development tool comprising SDKs and APIs, for example NVIDIA CUDA [Cud], AMD APP [Amd], OpenCL [Mun] and Microsoft DirectCompute [Dir], facilitates the porting of algorithms on multi-core hardware.

GPUs for general-purpose computing is rather a new buzzword. It takes advantage of a programming paradigm, referred to as stream processing, which takes simple data repeatedly

and performs small computes operations with almost the same cost. The compute model is quite interesting to speed-up computer vision algorithms. It is still an emerging area for use in computer vision applications. Nevertheless some noteworthy performance increases are achieved, and hence it is worthwhile to explore the area. In the literature, many computer vision algorithms are implemented on different parallel architectures and programming models, both by academia and industry in the form of articles and white papers. In this section, we summarize these different contributions.

To take advantage of stream processing the algorithm does requires some modifications to harness the maximum of the available compute power. As the model is not versatile, it might not result in desirable speedups for some applications, like signal processing. In this study, we focus on the use of GPUs for stream processing, due to its compute performance and off-the-shelf availability. A range of supporting tools and libraries make the learning curve simpler and development time shorter. Before the days of toolkits for using GPUs for general-purpose computing, the understanding of the graphics pipeline and shader programming languages was essential. Now the toolkits provide an interface similar to C, which enables the porting of a wide variety of algorithms with ease. The supporting tools do facilitate programming for graphics processors, but it requires the developed programs to follow the guidelines defined by the programing model and the memory model, in order to achieve maximum utilization of processing capabilities of the hardware. All details regarding the hardware specifications, the programming model and the memory model are detailed in Appendix D.

One of the biggest challenges in optimizing GPU code for data dominated applications is the management of the memory accesses, which is a key performance bottleneck. Memory latency can be several hundreds even thousands of clock cycles. This can be improved first by memory coalescing when memory accesses of different threads are consecutive in memory addresses. This allows the global memory controllers to execute burst memory accesses. The second optimization is to avoid such memory accesses through the use of a cache memory, internal shared memories, or registers. Most importantly, shared memory is an on-chip high bandwidth memory shared among all the threads on a single SM. It has several uses: as a register extension, to avoid global memory accesses, to give fast communication among threads, and as a data cache.

Different studies have implemented visual saliency models on GPUs. For instance, GPU implementation [HZ07] of visual saliency model using static features (color, intensities and orientations), achieved speedups up to *53 fps* for image sizes 1280×720. The model is based on Itti and VOCUS models to extract salient regions. Similarly, [Hil+10] processed Itti's visual saliency model for 512×512 images in $22ms$ using a single GPU. There is a significant improvement in the model's computation times as the original iLab's implementation of visual saliency takes around $52ms$. A two-GPU implementation [PS10] of Itti's model performed the computation of saliency at *30 fps*. The implementation used *two GPUs* generating 72 filter and 42 feature maps in total. A more powerful implementation by [Xu+09] achieved *313 fps* for computation of 640×480 visual saliency maps using *four GPUs*.

The chapter is organized as follows: in Section 5.2, we describe the implementation of the visual saliency model. We present several important kernels in detail, and discuss different optimizations done to improve speedups. Section 5.4 reports the speedups, and also evaluates the validity of these results. In the end, Section 5.5 concludes the chapter.

5.2 Saliency model on GPUs

Before porting an application onto CUDA-enabled devices, it is important to take into consideration two points. First, the application ported will get the desired performance increase. Second, the sequential code can decomposed into two sub-programs: one running on the host (the CPU) and other running on the device (the GPU). In general, the host is responsible for all data read/write operations from disk and data copy operations to/from the device, while the devices performs the task in parallel.

5.2.1 The static pathway

To start with the mapping of the static pathway of the visual saliency model on a GPU, it is partitioned into data-parallel portions of code, isolated as separate kernels. On main program execution, the host declares all variable on the host memory as well as on the device memory. Some of these variables are initialized, and bound to read-only memories on the device, such as constant and texture memory. On program iteration, the host fetches an input video frame from the main memory, which is transferred to the device memory. The data is used by data-parallel kernels, executed sequentially by the host.

The sequentially executed kernels are shown in Algorithm 2. First, a couple of preprocessing steps are performed on the visual input, to give it more detail. The steps include applying the Retinal filter and the Hanning mask. Then, in the frequency domain, the data is treated with a 2D Gabor filter bank with *six orientations* and *four frequency bands*; resulting in *24 partial maps*. After moving back to the spatial domain, interactions among the neighboring partial maps are applied. They are inhibited or excited depending on the maps' orientation and frequency band. The resulting partial maps are normalized, and finally accumulated into a saliency map M^s for the static pathway.

Algorithm 2 Static pathway of visual saliency model

Precondition: An image X of size $W \times H$

```
1 function StaticPathway(X)
2     X' ← RetinalFilter(X)
3     X'' ← HanningMask(X')
4     M_{u,v} ← GaborFilterBank(X'')
5     M_{u,v} ← Interactions(M_{u,v})
6     M_{u,v} ← Normalizations(M_{u,v})
7     M^s ← Fusion(M_{u,v})
8     return M^s
```

The static pathway includes the Retina filter with low-pass filters using 2D convolutions and recursive Gaussian filters, normalizations with reduction operations, some simple matrix operations and Fourier transforms. Here, the complex Fourier transformations are carried out using the NVIDIA CUDA fast Fourier transform library (cuFFT) that provides a simple interface for computing FFTs up to 10× faster. The reductions use Thrust library, an interface to many GPU algorithms and data structures. In this section, we present two kernels, the Gabor filtering and the interactions kernel, as they are improved using several optimizations to the kernel code, and some adjustments to their kernel launch configurations.

5.2.2 The dynamic pathway

Similar to the implementation of the static pathway, we first perform task distribution of the algorithm, and realize its sequential version. Some of the functional units are: recursive Gaussian filter, Gabor filter bank to break the image into sub-bands of different orientations (*six orientations* and *three frequencies*), Biweight Tuckey motion estimator, Gaussian pre-filtering for pyramids, spatial and temporal gradient maps for estimation, and bilinear interpolation. After testing these functions separately, they are put together to give a complete sequential code. The algorithm being intrinsically parallel allows it to be easily ported to CUDA.

Algorithm 3 describes the dynamic pathway, where first camera motion compensation and retinal filtering is done as a preprocessing on the visual input. Afterward, the preprocessed input is passed onto the motion estimator implemented using 3^{rd} order Gabor filter banks. The resulting motion vectors are normalized using temporal information to get a dynamic saliency map.

Algorithm 3 Dynamic pathway of visual saliency model

Precondition: An image X of size $W \times H$

```
1 function DynamicPathway(X)
2     X^c ← MotionCompensation(X)
3     P ← CreatePyramid(X^c)
4     P' ← RetinalFilter(P)
5     V ← MotionEstimation(P')
6     M^d ← TemporalFilter(V)
7     return M^d
```

5.2.3 The face pathway

The Algorithm 4 describes the face pathway, where the input image on the device is first preprocessed using retinal filtering. OpenCV's face detection on the GPU is used to detect the faces. The resulting faces are copied back to the host, where they are post-processed to discard the bad detections. In the end, a face saliency map is built using these filtered faces.

Algorithm 4 Face pathway of visual saliency model

Precondition: An image X of size $W \times H$

```
1 function FacePathway(X)
2     X' ← RetinalFilter(X)
3     faces ← FaceDetector(X')
4     faces_p ← Postprocessing(faces)
5     M^f ← ComputeFaceMap(faces_p)
6     return M^f
```

Ultimately, the saliency maps from the static, dynamic and face pathways are copied back onto the host CPU, where they are fused together outputting a saliency map. The saliency maps from different pathways, and the final output saliency map is shown in Figure 5.1.

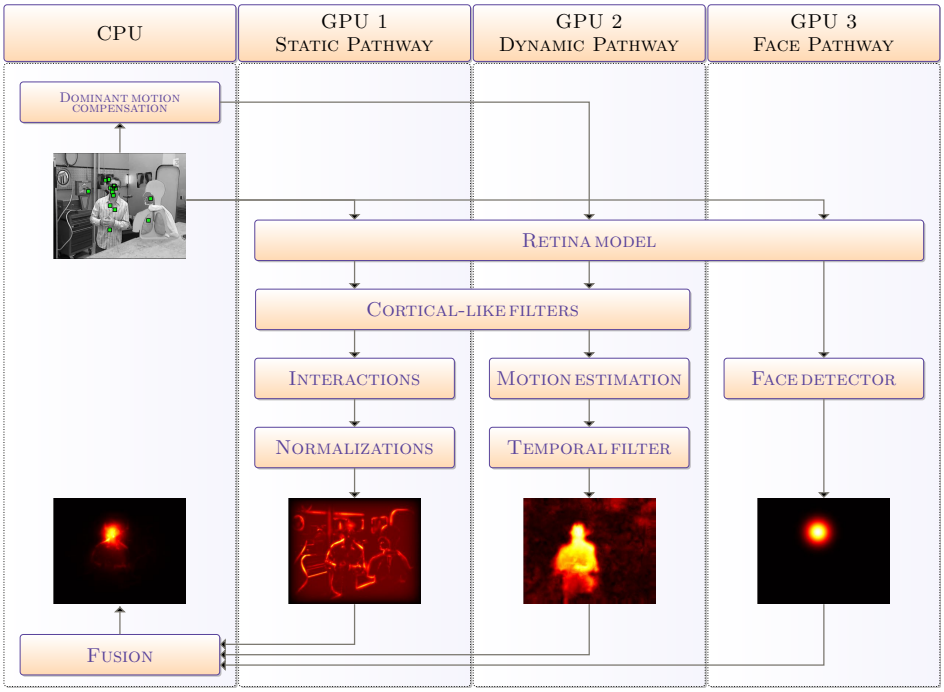


Figure 5.1: Parallel implementation of the visual saliency model.

5.3 Sample kernels from the implementation

5.3.1 Retinal filter

The visual information is prefiltered in a manner similar to one in the retina, where it is transfered all the way from the photoreceptor cells to the bipolar cells. The filter is referred to as a *'retinal filter'* (Figure 5.2), modeled by Gaussian low-pass filtering to remove higher frequencies, to get a smoother energy spectrum. Their difference at the end to reduce the lower frequencies acts as a band-pass filter. The result is visual information with more details.

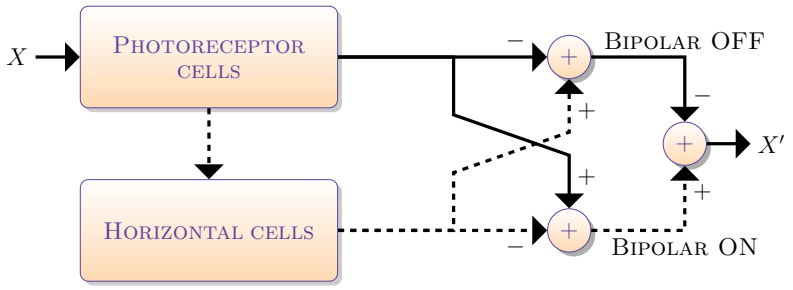


Figure 5.2: Retinal filtering.

Algorithm 5 shows the visual information being processed by Photoreceptor (Figure 5.3a) and Horizontal cells (Figure 5.3b), both modeled as low-pass filters. Afterward, the state of the bipolar cells determines the output from parvocellular (Figure 5.3c) or magnocellular cells (Figure 5.3d). The former with more details compared to the latter. Each output is

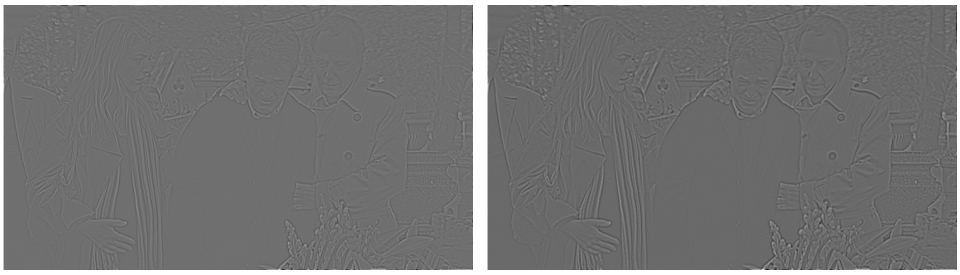
used by the ventral stream (the static pathway) and the dorsal stream (the dynamic pathway) respectively.

Algorithm 5 Pseudocode for Retinal filtering

Precondition: An image X of size $W \times H$

```
1 function RetinalFilter(X)
2     P ← X * g_e        ▷ g_e: Gaussian envelope for low-pass filter, the Photoreceptor cells.
3     H ← P * g_e        ▷ g_e: Gaussian envelope for low-pass filter, the Horizontal cells.
4     B^on ← P − H
5     B^off ← H − P
6     X' ← B^on − B^off  ▷ Difference of bipolar cells states, 'on' and 'off'.
7     return X'
```

(a) Photoreceptor cells. **(b)** Horizontal cells.



(c) Parvocellular cells. **(d)** Magnocellular cells.

Figure 5.3: Outputs from Retinal filtering.

5.3.2 Gabor filter bank

The visual information processed in different frequencies and orientations in the primary cortex can be modeled as a Gabor filter bank. Here, we present the bank for the static pathway. It uses *six orientations* and *four frequencies* to obtain *24 partial maps*, as illustrated in Figure 5.4.

Algorithm 6 shows the Gabor bank applied to the preprocessed visual input in frequency domain. The resulting *24 partial maps* $M_{u,v}$ are transformed to spatial domain. The transformations from spatial to frequency domain and back to spatial from frequency domain are done using the cuFFT library for GPUs.

The input data is in frequency domain, represented as complex numbers. We define a thread block of *32 threads*, where each thread fetches one number from the pair, either the real part or the imaginary part depending on its thread id. For instance in Figure 5.5,

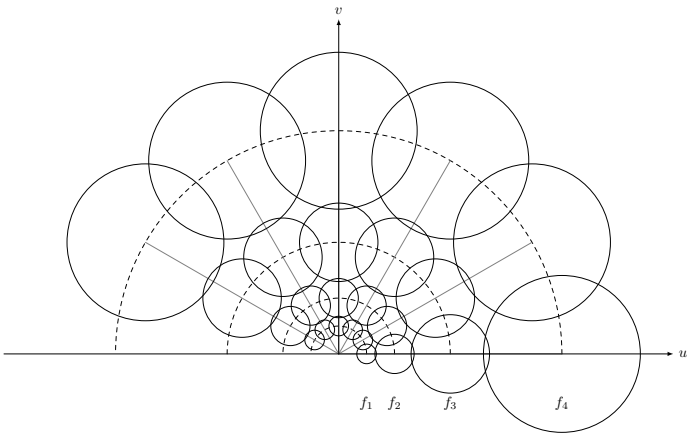


Figure 5.4: Gabor filter bank configuration in the plane (u,v) with six orientations (0°, 30°, 60°, 90°, 120°, 150°) and four frequency bands ($f_1 = 0.03125$, $f_2 = 0.0625$, $f_3 = 0.125$, $f_4 = 0.25$).

Algorithm 6 Static pathway of visual saliency model

function GaborFilterBank(Y)
  $\widehat{Y} \leftarrow \mathcal{F}(Y)$
  for all $u \in$ orientations **do**
    for all $v \in$ frequencies **do**
      $\widehat{M}_{u,v} \leftarrow \widehat{M}_{u,v} \cdot g_{u,v}$ ▷ $g_{u,v}$: Gabor function at (u,v)
      $M_{u,v} \leftarrow \mathcal{F}^{-1}(\widehat{M}_{u,v})$
  return M

thread 0 fetches the real part of the first complex number. 24 Gabor functions are applied to the fetched number, and the results are stored to the device memory. The Gabor functions applied correspond to six orientations and four frequencies. They are precomputed on the host, and are bound to the read-only texture memory on the device. This step is done once before the host initiates the execution of kernels.

5.3.3 Interactions between neighboring maps

Block diagram in Figure 5.6 shows the interactions kernel for 32 complex numbers for *24 partial maps* from Gabor filter bank that are first accessed from the global memory in coalesced manner, and placed into shared memory into separate regions for real and complex portions of the number. This data is then accessed in a random manner, and converted into 16 real numbers. The results are stored in shared memory variable, which is afterward used as prefetched data for the next phase of neuronal interactions.

Listing 5.1: Interactions kernel in static pathway

```
__global__ void ShortInteractionKernel ( Complex* in, unsigned int width
                  ,unsigned int height, float* out) {
  __shared__ float maps[ NO_OF_ORIENTS * NO_OF_BANDS ][32];
  __shared__ float buf [72];

  unsigned int x1 = blockIdx.x*blockDim.x + threadIdx.x/2;
  unsigned int x2 = blockIdx.x*blockDim.x + threadIdx.x;
```

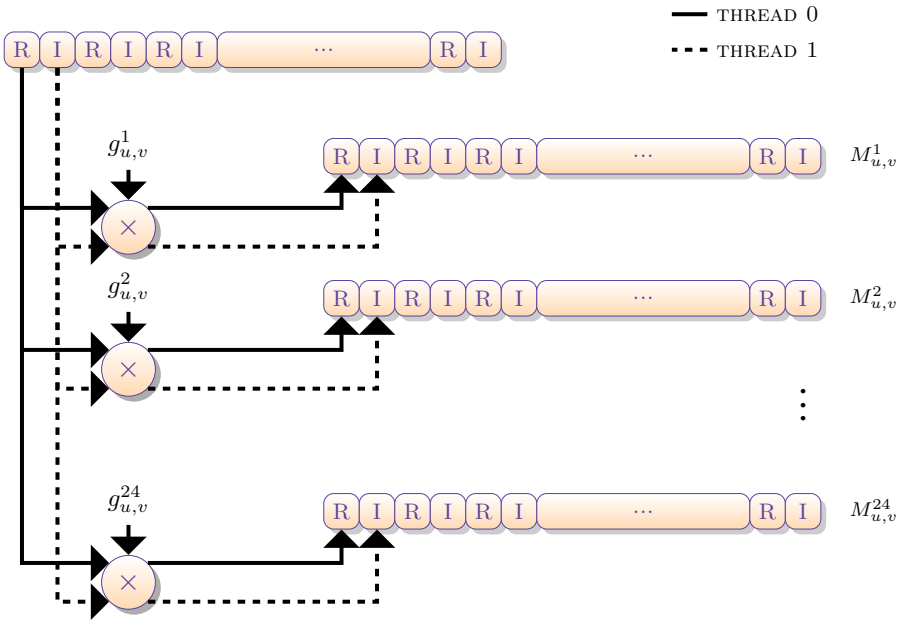


Figure 5.5: Two sample threads from a thread block for Gabor filter bank.

```
8   unsigned int y  = blockIdx.y*blockDim.y + threadIdx.y;
9
10  if ( x1 >= width || x2 >= width || y>= height) return;
11
12  unsigned int mod  = threadIdx.x1%2;
13  unsigned int pt   = threadIdx.x1/2 + 40* mod;
14  unsigned int size = width*height;
15
16  for ( unsigned int j=0 ; j< NO_OF_ORIENTS ; ++j) {
17    for ( unsigned int i=0 ; i< NO_OF_BANDS ; ++i) {
18      /* ************************************************************
19      * 32 threads process 16 complex numbers in parallel
20      * every thread stores them with real and imaginary interlaced
21      * 32 threads produce 32 real products in parallel
22      ************************************************************ */
23      buf[pt] = // first 16 complex numbers
24        in[(j* NO_OF_BANDS+i)*size+(y*width+x1)][mod]/(float)(size);
25      buf[pt+16] = // next 16 complex numbers
26        in[(j*NO_OF_BANDS+i)*size+(y*width+x1+16)][mod]/(float)(size);
27      __syncthreads();
28
29      maps[j* NO_OF_BANDS + i][ threadIdx.x] = abs(
30        buf[threadIdx.x ]*buf[threadIdx.x ] +
31        buf[threadIdx.x + 40]* buf[threadIdx.x + 40]);
32      __syncthreads ();
33    }
34  }
35  // prefetched data in shared memory is used by interactions
36 }
```

❖ **Coalescing using shared memory** A method to avoid non-coalesced memory accesses is by re-ordering the data in shared memory. To demonstrate the uses of shared memory, we take an example kernel as shown in Listing 5.1, which is illustrated using a block diagram shown in Figure 5.6. Here, the data values are in a complex format consisting of two floats. The very first step is fetching the values into shared memory, where each float is read by a separate thread as shown in Line 12. These global memory

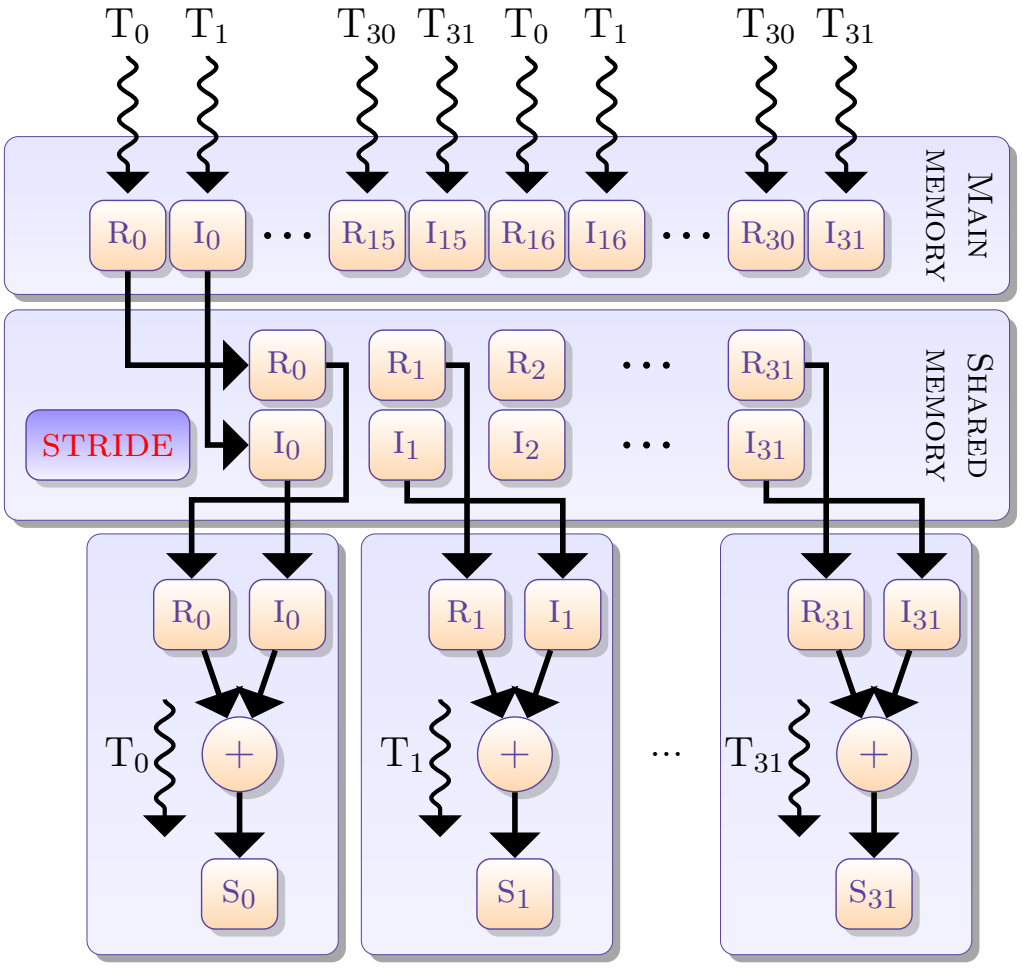


Figure 5.6: Block diagram of data-parallel interaction kernel

accesses are coalesced, as contiguous floats are read shown in Line 24. Furthermore, we use two shared buffers; one for real part, and the other for imaginary part. This arrangement gives uncoalesced shared memory accesses during computation in Line 31 to convert the complex numbers into real, and also to scale down the output from the unnormalized Fourier transforms done using cuFFT library.

	GPU time	Occupancy	Shared memory per block (bytes)
Uncoalesced memory accesses	38ms	0.125	0
Coalesced memory accesses	12ms	0.125	288

Table 5.1: Profile of interaction kernel for memory coalescing using shared memory.

❖ **Avoiding shared memory bank conflicts** In case if multiple threads access the same bank causes conflicts. These conflicting accesses are required to be serialized either using an explicit stride based on thread's ID or by allocating more shared memory. In our case, when thread's ID is used to access shared memory then a conflict occurs, as *thread 0* and *thread 1* access the same bank. Thus, we use a stride of 8 to avoid any conflicts as shown in Line 13. Although, a multiprocessor takes only *4 clock cycles* doing a shared memory transaction for entire half warp, but bank conflicts in the shared memory can degrade the overall performance.

	GPU time	Occupancy	Warp serialize
With bank conflicts	8.13ms	0.125	75529
Without bank conflicts	7.9ms	0.125	0

Table 5.2: Profile of interaction kernel for bank conflicts.

❖ **Prefetching using shared memory** Another use of shared memory is to prefetch data from global memory, and cache it. In the example kernel, the data after conversion and rescaling are cached as shown in Line 29, and this prefetched data is used in the next phase of interactions.

5.3.4 Gaussian recursive filter

The Gaussian filtering can be approximated using a recursive filter. The result is reduced complexity of Gaussian filtering. [YV95] propose a technique to obtain recursive filter of order N that can give a best approximate of the Gaussian envelope. The transfer function used is decomposed into two passes: causal and anti-causal (Figure 5.7).

$$H(z) = H^{+}(z) \cdot H^{-}(z)$$

Here, the casual H^{+} and anti-causal H^{-} passes are computed using the equations.

$$v[n] = \alpha x[n] - \sum_{i=1}^{N} b_i v[n-i]$$

$$y[n] = \alpha v[n] - \sum_{i=1}^{N} b_i y[n+i]$$

where $x[n]$ is the input signal and $y[n]$ is the output signal. The coefficients α and b_i are calculated using width of the filters using formulas described in [YV95]. The anti-causal pass is shown in Listing 5.2.

The advantage of using Gaussian recursive filtering is that the per pixel operations remains constant regardless of the filter used. Only the computation times depend on the order of the recursive filter. Here, 3^{rd} order recursive filters are used to give a good approximation of Gaussian filtering compared to 1^{st} order recursive filters. The difference between the two is an increased computation time for the later case.

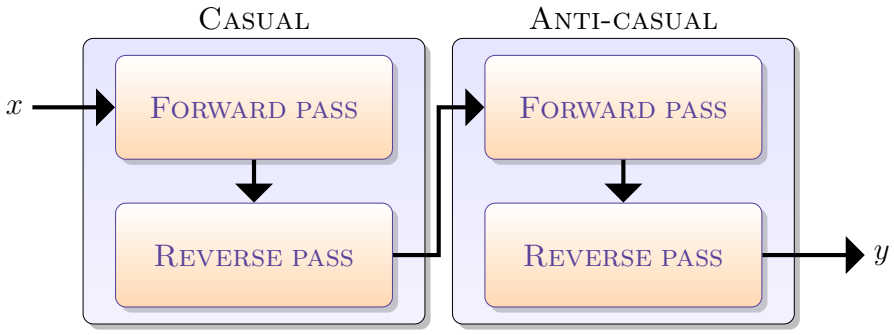


Figure 5.7: Recursive implementation of Gaussian filter decomposed into two passes: causal and anti-causal.

Listing 5.2: Anti-casual Gaussian recursive filter kernel

```
__global__ void Dynamic::KernelGaussianRecursiveAntiCausal( float *odat, float *idat, size_t im_size, float B, float b0, float b1, float b2, float b3)
{
    float yp1, yp2, yp3, xc, yc;

        int x = threadIdx.x + blockIdx.x*blockDim.x;
        if( x>=im_size.w) return;

        idat += ( x + blockIdx.y * (im_size.w*im_size.h));
        odat += ( x + blockIdx.y * (im_size.w*im_size.h));

    /**************
    * Forward pass
    **************/
        yp3 = B*( *idat);
        yp2 = B*( *(idat + im_size.w)) + ( b1/b0) * yp3;
        yp1 = B*( *(idat + im_size.w)) + ( b1*yp2 + b2*yp3) / b0;

        for( int y=3 ; y<10 ; y++){
                yc = B*( *(idat + y*im_size.w)) + ( b1*yp1 + b2*yp2 + b3*yp3) / b0;

                yp3 = yp2; yp2 = yp1; yp1 = yc;
        }

        for( int y=0; y<im_size.h ; y++){
                xc = *idat;
                yc = B*xc + ( b1*yp1 + b2*yp2 + b3*yp3) / b0;

                yp3 = yp2; yp2 = yp1; yp1 = yc;
                *odat = yc;
                idat += im_size.w; odat += im_size.w;
        }
        // reset pointer
        idat -= im_size.w; odat -= im_size.w;

    /**************
    * Reverse pass
    **************/
        yp3 = B*( *idat);
        yp2 = B*( *(idat - im_size.w)) + ( b1/b0) * yp3;
        yp1 = B*( *(idat - im_size.w)) + ( b1*yp2 + b2*yp3) / b0;

        for( int y=3 ; y<10 ; y++){
                yc = B*( *(idat - y*im_size.w)) + ( b1*yp1 + b2*yp2 + b3*yp3) / b0;

                yp3 = yp2; yp2 = yp1; yp1 = yc;
        }
```

```
47
48          for( int y = im_size.h-1; y >= 0; y--) {
49                  xc = *idat;
50                  yc = B*xc + ( b1*yp1 + b2*yp2 + b3*yp3) / b0;
51
52                  yp3 = yp2; yp2 = yp1; yp1 = yc;
53                  *odat = yc;
54                  idat -= im_size.w; odat -= im_size.w;
55          }
56  }
```

5.3.5 Motion Estimator

The motion estimator [BP02] presented here employs differential method using Gabor filters to estimate local motion. The estimated speeds are obtained by solving a robust system of equations of optical flow. It works on the principle of conservation of brightness, that is the luminance of any pixel remains the same for a certain interval of time. Considering that $I(x,y,t)$ represents the brightness function for the sequence of images then according to the hypothesis of conversation of total luminance, its time derivative is zero. Therefore, the motion vector $v(p) = (v_x, v_y)$ can be found using the equation of optical flow:

$$\frac{\mathrm{d}I(p,t)}{\mathrm{d}t} = \nabla I(p,t) \cdot v(p) + \frac{\partial I(p,t)}{\partial t} = 0$$

where $\nabla I(x,y,t)$ is the spatial gradient of luminance $I(x,y,t)$. Using this equation, we get a velocity component parallel to this spatial gradient. The information from the spatial gradients in different directions can be used to find the actual movement of an object. If the intensity gradient is $\nabla l = 0$ then the motion is negligible, whereas motion in one direction represents the edges.

To perform a proper estimation of motion, the movement is averaged corresponding to its spatial neighborhood. This spatial neighborhood is required to be large enough to avoid any ambiguous resulting motion vectors. Thus, to get a spatial continuity within the optical flow, we convolve the spatio-temporal image sequence with a Gabor filter bank:

$$v_x \cdot \frac{\partial (I * G_i)}{\partial_x} + v_y \cdot \frac{\partial (I * G_i)}{\partial_y} + \frac{\partial (I * G_i)}{\partial_t} = 0$$

the bank consists of N filters G_i with the same radial frequency. The result is a system of N equations for each pixel with velocity vector composed of two components (v_x, v_y). This system of equations can be represented as:

$$\begin{pmatrix} \Omega_2^x & \Omega_1^y \\ \Omega_2^x & \Omega_2^y \\ \cdots & \\ \Omega_n^x & \Omega_n^y \end{pmatrix} \cdot \begin{pmatrix} v_x \\ v_y \end{pmatrix} = \begin{pmatrix} \Omega_2^t \\ \Omega_2^t \\ \cdots \\ \Omega_n^t \end{pmatrix}$$

this over-determined system of equations is solved the method of least squares (Biweight Tuckey test) [BP02].

The estimator uses multi-resolution patterns to allow robust motion estimation over a wide range of speeds. Here, the image is sub-sampled into multiple scales resulting in a pyramid, where the approximation begins with the sub-sampled version of the image at the highest level. This process is iterated until applied to the image with original resolution. This

multi-scale approach is equivalent to applying a Gabor bank of several rings with different scales. The final result for the estimator is a motion vector for each pixel.

Listing 5.3: Motion estimator kernel

```
1  texture<float, 1, cudaReadModeElementType> tex_dx, tex_dy, tex_wi;
2  __global__ void DYN_ker_motion_estimator( float *xv, float *vy, int w, int h)
3  {
4    unsigned int i, j, m;
5    float _wi, _ri;
6    __shared__ float _dx, _dy, Mx, My, W, mxp, myp;
7
8    m = threadIdx.x + blockIdx.x*blockDim.x; if( m>=(w*h)) return;
9
10   for( j=0 ; j<NO_OF_STEPS ; ++j){
11     mxp[threadIdx.x] = Mx[threadIdx.x]; myp[threadIdx.x] = My[threadIdx.x];
12     Mx[threadIdx.x] = 0; My[threadIdx.x] = 0; W[threadIdx.x] = 0;
13     syncthreads();
14
15     for( i=0 ; i<N*(2*N-1) ; ++i){
16       _dx[threadIdx.x] = tex1Dfetch( tex_dx, m + i*(w*h));
17       _dy[threadIdx.x] = tex1Dfetch( tex_dy, m + i*(w*h));
18       _wi        = tex1Dfetch( tex_wi, m + i*(w*h));
19       syncthreads();
20
21       _ri = sqrtf( powf( _dx[threadIdx.x]-mxp[threadIdx.x],2) +
22              powf( _dy[threadIdx.x]-myp[threadIdx.x],2));
23
24       if( fabsf(_ri)<C && _wi!=0){ _wi=( _ri*_ri - C*C)/(C*C); _wi*=_wi;}
25
26       Mx[threadIdx.x]+=_wi*_dx[threadIdx.x];
27       My[threadIdx.x]+=_wi*_dy[threadIdx.x];
28        W[threadIdx.x]+=_wi;
29       syncthreads();
30     }
31     if ( W[threadIdx.x]!=0){
32       Mx[threadIdx.x]/=W[threadIdx.x]; My[threadIdx.x]/=W[threadIdx.x];
33     }
34     syncthreads();
35   }
36   vx[m] += Mx[threadIdx.x]; vy[m] += My[threadIdx.x];
37 }
```

- **Reducing device memory accesses** In motion estimator prefetch kernel, we use spatial and temporal gradient values to get $N(2N-1)$ solutions that are used to perform iterative weighted least-square estimations. These numerous intermediate values are stored as array variables because the register count is already high. Consequently, this leads to costly global memory accesses. The accesses can be avoided by placing some values in shared memory, to get a solution with the fewer memory accesses, and efficient use of the limited resources on the device. We achieved performance gains by carefully selecting the amount of shared memory without compromising the optimal number of active blocks residing on each SM.

- **Reducing register count** In the motion estimator kernel, there is a limitation of higher register count due to the complexity of the algorithm; hence, resulting in a reduced number of active thread blocks per SM. In our naive solution, the register count is 22 that can be considerably reduced to 15 registers per thread block using shared memory for some local variables, as shown in Line 6 in Listing 5.3. Consequently, the occupancy increased from 0.33 to 0.67 with 8 active thread blocks residing on each

SM. The variables to be placed in shared memory are carefully selected to reduce the number of synchronization barriers needed.

❖ **Using texture memory** Texture memory provides an alternative path to device memory, which is faster. This is because of specialized on-chip texture units with internal memory to allow buffering of data from device memory. It can be very useful to reduce the penalty incurred for nearly coalesced accesses.

In our implementation, the main motion estimation kernel exhibits a pattern that requires the data to be loaded from device memory multiple times i.e. we calculate a $N \times (2N-1)$ system of equations for $2 \times N$ levels of spatial and temporal gradients. Due to memory limitations of the device, it is not feasible to keep the intermediate values. So, we calculate these values at every pass of the estimator leading to performance degradation because of higher device memory latency. As a solution, the estimation kernel can be divided into two parts: one calculating the system of equations, while other using these equations for estimation through texture memory. Here, in lrefestimator, texture memory's caching mechanism is employed to prefetch the data calculated in previous kernel, hence, reducing global memory latency and giving up to 10% performance improvement.

	No. of registers	Occupancy	GPU time %
Naive solution	18	0.75	32
Shared memory	29	0.50	31.5
Shared + prefetching	32	0.50	31.7
Shared + textures	17	0.375	32

Table 5.3: Profile of motion estimator kernel.

5.4 Experimental results

5.4.1 Performance analysis

All implementations are tested on a *2.80 GHz* system with *10 GB* of main memory, and Windows 7 running on it. On the other hand, the parallel version is implemented using latest *CUDA v4.2* programming environment on *NVIDIA GeForce GTX 285* series graphics cards.

Speedup of static pathway

In the implementation, a static saliency map M^s is produced at the end of the static pathway, which identifies the salient regions in the visual input. The different stages of the pathway include: a Hanning mask, a retinal filter, a Gabor filter bank, an interaction function, some normalizations and a final summation. All these stages show a great potential to be parallelized, thus, they are isolated within separate kernels. Initially, the model is implemented using MATLAB that happens to be extremely slow taking around 34*s* (Table

5.8) to compute the saliency maps for videos with frame sizes 640×480 pixels. We improve the pathway by porting it to C. It includes many optimizations and also the use of highly optimized FFTW library for Fourier transforms, but the speedup witnessed is only 2.17×. We get further improvement using multithreading up to a factor of 1.66 over sequential C implementation.

An advantage of the C implementation is to identify the data-parallel portions, as well as to write the program in a familiar language. The data-parallel portions are implemented as separate CUDA kernels for GPU code. This resulted in 149× speedup over multithreaded implementation, just after partitioning into data-parallel portions. However, to achieve the promised speedup the code requires many tweaks and optimizations, which happens to be a complex maneuver. The peak performance topped over to 166× after making various optimizations. Table 5.4 shows timings for the different kernels in the static pathway, while the performance gains after optimizations are presented in Table 5.5.

Kernel	Duration (ms)
Mask	0.08
FFT	0.59
Shift	0.09
Gabor	1.47
Inverse shift	1.13
IFFT	10.76
Interaction	3.13
Normalize	3.33
Normalize Itti	3.34
Normalize Fusion	2.89
Total	26.81

Table 5.4: Computational cost of each step in static pathway for video with frame size 640×480 on NVIDIA GeForce GTX 285. The results are obtained using CUDA visual profiler.

Case	Over MATLAB	Over First CUDA	MPixels/sec
First Implementation	149×	1.00×	4.11
Textures used	158×	1.06×	5.93
No bank conflicts	161×	1.08×	6.08
Fast math used	166×	1.11×	6.25

Table 5.5: Speedups after optimizations to the GPU implementation of static pathway against the multithreaded C implementation.

The algorithm implemented in CUDA is ported from MATLAB code, where all the computations are done entirely in double-precision. Fortunately, the effects of low-precision in parallel implementation are not obvious. The main reason is the type of algorithm, whether

it can produce acceptable results, or ones that are usable. Here the resulting saliency map may be inaccurate, but visually fine with universal image quality index [WB02] of 99.66% and 2-digit precision among the 24-bits of a float mantissa. Figure 5.8 shows the mean error with respect to the reference during different stages of the pathway. We observe that the accuracy increases along the progressing stages because of the reduction of information, more evident during Gabor filtering and normalization phases until finally ending up in regions that are salient.

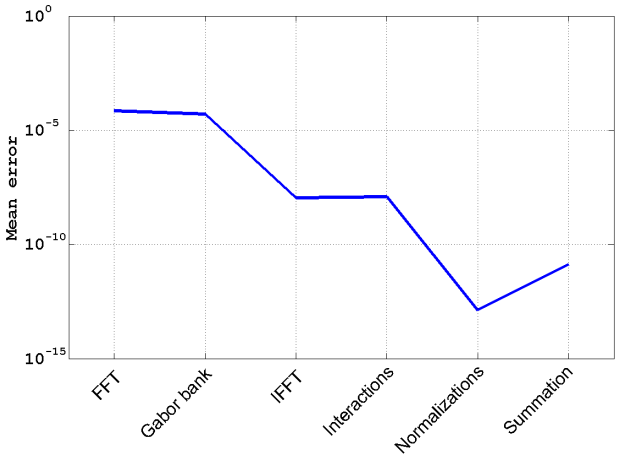


Figure 5.8: Mean error with respect to the reference MATLAB implementation of static pathway, to determine the impact of lower precision on GPUs.

Speedup of dynamic pathway

In the implementation, a dynamic saliency map M^d is produced at the end of the dynamic pathway, which identifies the salient regions in the visual input based on motion. The different stages of the pathway include: a retinal filter, a Gabor filter bank, a motion estimator and a temporal filter. The first implementation in MATLAB takes around 237s (Table 5.8) to compute the saliency maps for videos with frame sizes 640×480 pixels. We improve the pathway by porting it to C. We get an improved performance of factor 7.6×. It is further improved by a multithreaded solution giving 1.41× improvement over the sequential C implementation. The final port to GPU results in 184× speedup over the multithreaded CPU implementation. Table 5.6 shows timings for the different kernels in the dynamic pathway.

To evaluate the correctness of the motion estimator, we calculate error between estimated and real optical flows using the equation below:

$$\alpha_e = \arccos\left(\frac{uu_r + vv_r + 1}{\sqrt{u^2+v^2+1}\sqrt{u_r^2+v_r^2+1}}\right)$$

where a_e is the angular error for a given pixel with (u, v) the estimated and (u_r, v_r) the real motion vectors. We used *'treetran'* and *'treediv'* image sequences for the evaluation, showing translational and divergent motion respectively [BP02]. The results illustrated in Table 5.7 are obtained using *'treetran'* and *'treediv'* image sequences of sizes 150×150 pixels.

Kernel	Duration (ms)
Retinal Filtering	9.6
Modulation	0.3
Demodulation	0.8
Interpolation	0.1
Projection	0.2
Gaussian recursive Causal	30.3
Gaussian recursive Anti-causal	20.5
Gradients	1.0
Motion estimator	27.1
Median filtering	0.1
memset	0.4
Total	90.4

Table 5.6: Computational cost of each step in dynamic pathway for video with frame size 640×480 on NVIDIA GeForce GTX 285. The results are obtained using CUDA visual profiler.

	treetran		treediv	
	$\bar{x}$	σ	$\bar{x}$	σ
MATLAB	1.63	5.27	6.06	8.22
C	1.10	0.99	4.15	2.69
CUDA	1.19	1.00	5.73	3.91

Table 5.7: Evaluating the estimator using angular error.

Speedup of face pathway

To evaluate the performance gains of the face pathway, we compare the timings for sequential and parallel face detection routines. Both routines are present in OpenCV library. The result for test videos with different frame sizes show that the CUDA implementation of the face detector gives speedups of 4 to 16 times on NVIDIA GeForce GTX 285 compared to the sequential implementation of face detection, as illustrated in Figure 5.9.

In the implementation, a face saliency map M^f is produced at the end of the face pathway, which identifies the salient regions in the visual input based on faces. The different stages of the pathway include: a retinal filter followed by a face detector. The first implementation in MATLAB takes around 6*s* to compute the saliency maps for videos with frame sizes 640×480 pixels. We improve the pathway by porting it to C. We get an improved performance of factor 2×. The pathway is slightly improved using multithreading, whereas the final GPU implementation results in 46× speedup over the multithreaded CPU implementation.

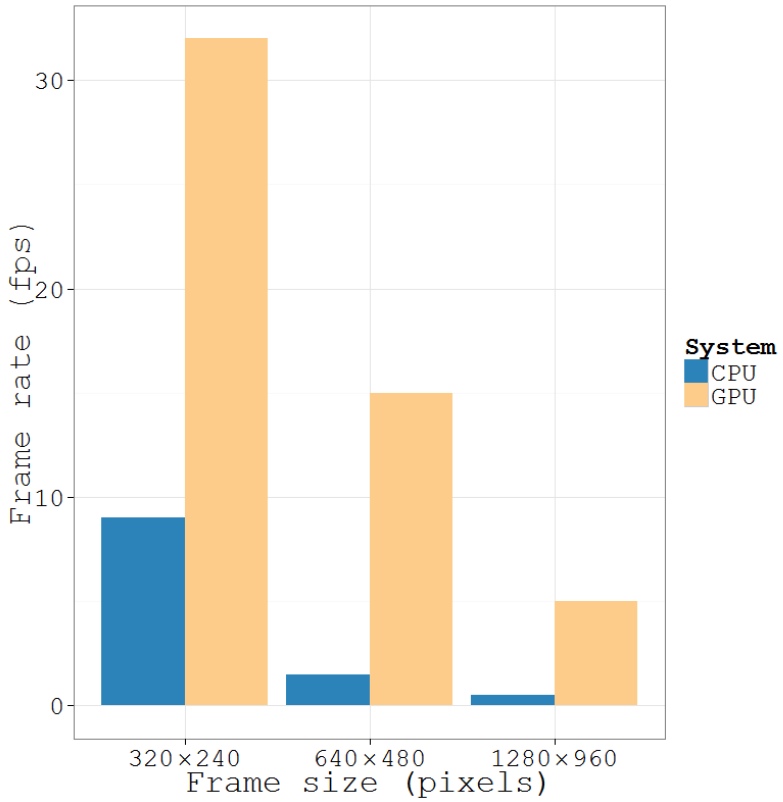


Figure 5.9: Timings of sequential and parallel face detection routines for video with frame size 640×480 on NVIDIA GeForce GTX 285.

5.4.2 Speedup over CPU version

The initial implementation of the visual saliency model was done in MATLAB. This implementation was first ported to sequential C implementation to get considerable improvement. The implementation now in a language, which is an interface to write CUDA programs. It was used to isolate all the data-parallel portions of the model, as CUDA kernels. The final GPU implementation gives incredible performance improvements compared to CPU implementations, as illustrated in Figure 5.10. The timings for all the different implementations are summarized in Table 5.8.

	M^s	M^d	M^f
MATLAB	34.01	237.03	6.18
C	10.73	31.24	3.26
C+OpenMP	6.65	22.13	3.21
CUDA	0.04	0.12	0.07

Table 5.8: Timings of various sequential and parallel implementations for video with frame size 640×480 on NVIDIA GeForce GTX 285.

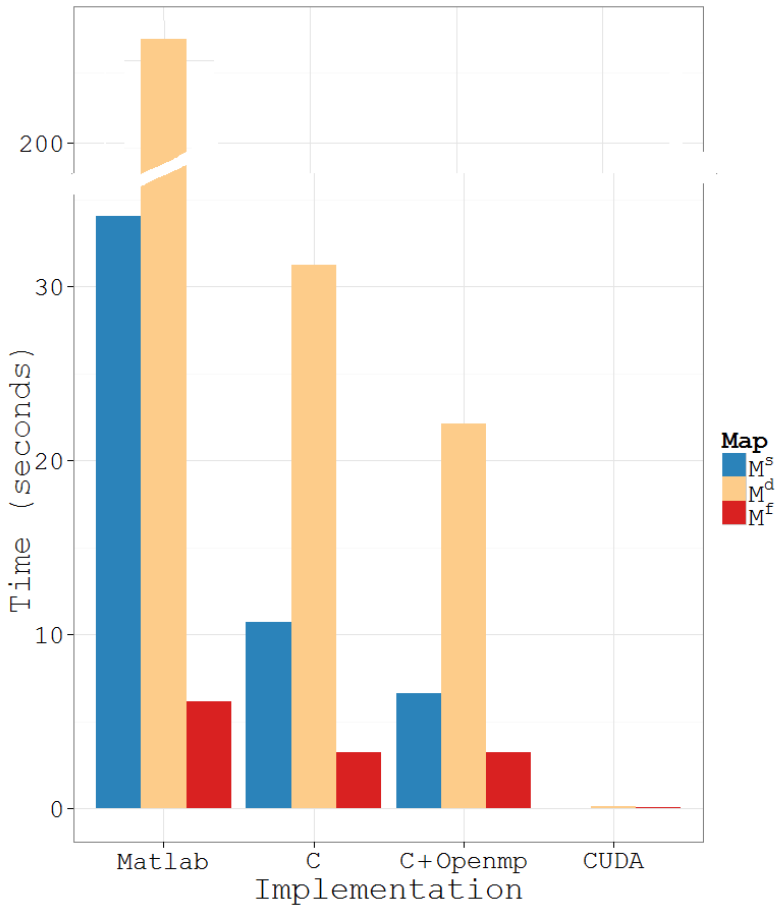


Figure 5.10: Timings of sequential and parallel implementations for video with frame size 640×480 on NVIDIA Geforce GTX 285.

5.4.3 Multi-GPU solution

Multi-GPU implementation is quite interesting to increase the computational efficiency of the entire visual saliency model. We have employed a shared-system GPU model, where multiple GPUs are installed on a single CPU. If the devices need to communicate, they do it through the CPU with no inter-GPU communication. A CPU thread is created to invoke kernel execution on a GPU, accordingly, we will have a CPU thread for each GPU. To successfully execute our single GPU solution on multi-GPUs, the parallel version must be deterministic. Our first implementation, the two pathways of the visual saliency model; static and dynamic, are completely separate with no inter-GPU communication required. The resulting saliency maps are simply copied-back to the host, where they can be fused together into the final saliency map.

Pipeline model

In this multi-GPU implementation, we employ simple domain decomposition technique by assigning separate portions of the task to different threads. As soon as any thread finds an available device, it fetches the input image from the RAM to device memory, and invokes the execution of the kernel. The threads wait until the execution of the kernel is complete, and then gather their respective results back.

In our implementation as illustrated in Figure 5.11, we have four threads that are assigned different portions of the visual saliency model. For instance: *thread 0* calculates the static saliency map, *thread 1* does the retinal filtering and applies the Gabor filters bank, *thread 2* performs the motion estimation outputting dynamic saliency map to the host, and *thread 3* calculates the face saliency map.

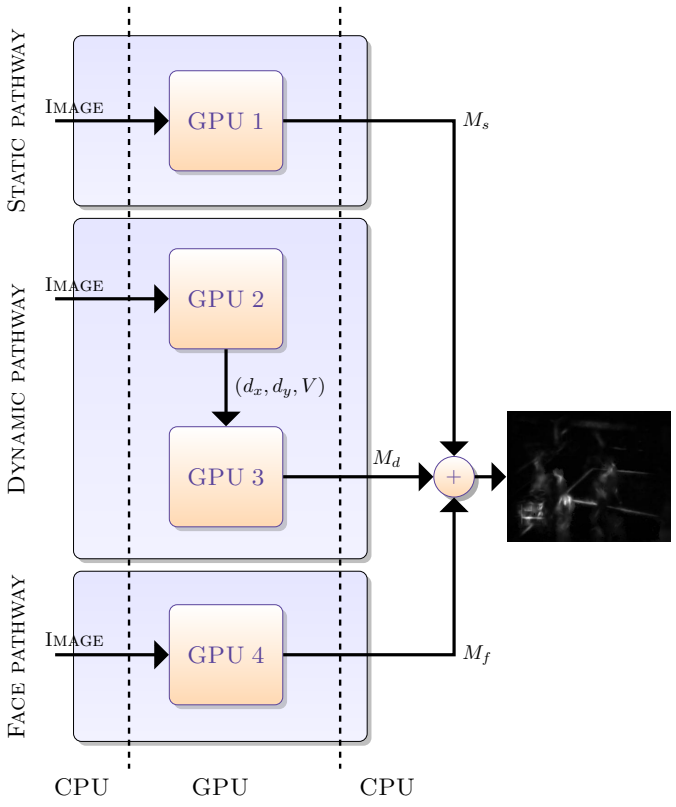


Figure 5.11: Block diagram of multi-GPU pipeline model

The division of the dynamic pathway is done based on computational times of different kernels, we find that the suitable cut will be after recursive Gaussian filters that are followed by the motion estimator as shown in Figure 5.12. Each half of this cut takes $\sim 50ms$, which is half of $90ms$ for entire pathway. The inter-GPU communication between *thread 1* and *thread 2* involves the transfer of N-level pyramid for the input image treated with the retinal filter and Gabor bank. Afterward, *thread 2* is responsible for the estimation. Finally, the static and dynamic saliency maps from *thread 0* and *thread 2* are fused together into the final visual saliency map. Consequently, using a pipeline cuts off the time to calculate the entire model to $\sim 50ms$ instead of $90ms$ for the simple solution.

Demonstrator

The program provides a method to compute visual saliency of a visual scene. It comprises of three separate pathways, each dedicated for a specific visual cue. The pathways are: static, dynamic and face pathway. All the specific saliency information from different pathways is combined or fused together to get the final or master saliency map. This saliency map shows the eye positions prediction made by the attention model. In Appendix E, we summarize

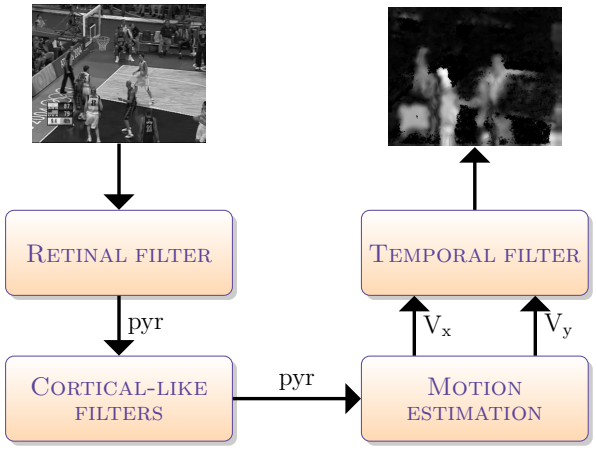


Figure 5.12: Block diagram of decompose dynamic pathway.

the different program options available, as well as, examples to demonstrate the use of the developed visual saliency model. The platform used is shown in Figure 5.13.

Figure 5.13: Platform for multi-GPU implementation of visual saliency model.

5.5 Summary and conclusion

In the chapter, we presented the multi-GPU implementation of a visual saliency model to identify the areas of attention. The main advantage of the performance gain accomplished will allow the inclusion of face recognition, stereo, audio, and other complex processes. Consequently, this real-time solution finds a wide application for several research and industrial problems, for example: video compression, video reframing, frame quality assessment, visual telepresence and surveillance, automatic target detection, robotics control, super-resolution, computer graphics rendering, and many more.

Science never solves a problem without creating ten more.

George Bernard Shaw

Chapter 6

Conclusions and perspectives

Studies conducted in this thesis focuses on faces and visual attention. We are interested to better understand the influence and perception of faces, to propose a visual saliency model with face features. Throughout the thesis, we concentrate on the question, "How people explore dynamic visual scenes, how the different visual features are modeled to mimic the eye movements of people, in particular, what is the influence of faces?". To answer these questions we use different steps: analyze the eye movements obtained from behavioral experiments, propose a model predicting them, and finally find a solution to make the model accessible to the research community.

In this thesis, we proposed a visual saliency model to determine the salient regions in videos, based on the observations from eye movement data. The data were acquired during free-viewing of videos. In conclusion, we have results that led to several contributions and their related perspectives.

6.1 Key contributions

In Chapter 3, we evaluated the preference of faces in videos. The study used data from a psycho-visual experiments to examine the gaze patterns of participants.

- Faces do attract attention in videos. We found that fixations on scene onset correspond to regions of interest in the previous scene, resulting in higher fixation dispersion among participants. As the scene progresses, dispersion decreases. It is much smaller for scenes with faces.
- Face region are salient in videos. The evaluation using different comparison criteria shows that fixations are made in proximity to the face regions. We conclude that this is essentially related to the informativeness and social significance of faces.
- Fixations are longer on faces in videos, in particular on scenes with only one face. We observe that initial fixations are shorter compared to following fixation, which seems likely to be influenced by center-looking strategy in the beginning, resulting in initial shorter fixations. The following fixations are longer, as they are made to extract maximum facial information. However, the durations are shorter, when several regions of interest, or two faces are competing for the limited attentional resources.

- ❖ We report that the preference of faces in dynamic scenes is influenced by different factors, such as the eccentricity, the area and the number of faces in the scene. We show that the influence of faces decrease with increasing eccentricities. It is relatively smaller for scenes with only one face compared to scenes with competing faces. In the later case, the factor of eccentricity from fixation is used to resolve competition between two faces. We confirm that an increase in the area of faces improve their visual performance by masking the effects the eccentricity and competition.

- ❖ We propose a modulation for individual faces based on the effects of eccentricity, area and number of faces in a scene. The modulation for faces is used for the proposed three pathway visual saliency model.

In Chapter 4, we use the observations about the influence of faces in videos to propose a face pathway.

- ❖ We propose a new bottom-up visual saliency model that breaks down the visual signal using three processing pathways based on different types of visual features: static, dynamic, and face. It is an extension of the saliency model proposed by [Mar+09]. The static and dynamic pathways are inspired by the biology of the first steps of the human visual system: a retina-like filter and a cortical-like bank of filters. The static pathway extracts the texture information based on luminance. The dynamic pathway extracts information about objects' motion against background. The face pathway extracts information about the presence of faces in the frames. The model also integrates center bias as a suitable modulation for the resulting visual saliency maps.

- ❖ We evaluated the proposed face pathway against eye movement data from the psycho-visual experiment. We show that the inclusion of face features improves the bottom-up visual saliency model.

- ❖ The eye movement experiment enables us to study which visual features attract a participant's gaze, and how to integrate them into the saliency model. It also helps us to devise an efficient and robust fusion of the three types of maps into a single master saliency map. Consequently, the coefficients selected for the three pathways, the maximum, the skewness and the confidence respectively, strengthen the most relevant feature maps in the master saliency map.

In Chapter 5, we present a multi-GPU implementation of a visual saliency model.

- ❖ The results show that the proposed GPU-based visual saliency model outperforms an equivalent CPU-based implementation by up to 132×. To our knowledge, this is the first GPU-based implementation of the model ever reported in the literature.

- ❖ Performance gains on GPUs can be obtained after careful consideration of thread and block configurations, memory allocations, and efficient global and shared memory accesses. The evaluation results for the optimized GPU program show higher efficiency compared to the unoptimized version of the program.

In conclusion, the main advantage of the gain in performance will allow the inclusion of other visual features and their faster evaluation against experimental eye positions data. Furthermore, the faster solution can be used in a wide variety of research and industrial problems.

6.2 Perspectives and future works

In light of the above, we can say that the original objectives of this research have been met. The following presents potential plans for future works, regarding faces in videos, saliency model and implementation:

- Haar features used by the face detector are not biologically-inspired. In this study, we used Viola-Jones face detector as a starting point because it is popular, fast, robust, and most importantly it has a parallel implementation. Consequently, the algorithm was used to propose a face pathway for the existing model with good detection results. The perspective is to propose a face pathway using some biologically-inspired face detection algorithms, such as spike neurons [VR+98] and Gabor filters [Phi+00].

- Studies have observed that using multi-modal stimuli can help make more accurate and faster eye movements during conditions of social interactions [Cor+02; AC03; Bel+11; Cou+12; SPG12; Vo+12]. For instance, during scenes with faces and sound, eye movements are directed to mouth regions to facilitate intelligibility of speech using lip-reading. In contrast, scenes without sound resulted in eye regions being fixated more often [Jef96; Vo+12]. Therefore, the inclusion of both audio and visual modalities into a saliency model can improve its performance. Also, it is more plausible to use such audio-visual model for scenes with sound, which are often encountered in real-life.

- A model with different low-level and high-level features require an efficient method to combine the information. Different fusion methods have their advantages and disadvantages depending on the video databases with varying attributes. A perspective is to define a robust fusion for all pathways of the model, which is a function of the application using the model.

- The proposed model is based on low-level visual features, it is 'bottom-up'. Its predictability can be increased by incorporating different high-level processes, such as the Working Memory (WM) and the Short-Term Memory (STM) in the visual system. Object perception is another example of high-level information with dorsal and ventral pathways interacting to complete object detection and recognition tasks. It will be interesting to incorporate some of this information into the existing model, to predict eye movements for longer scenes with or with any task.

- We used video databases comprising short video excerpts to evaluate the proposed 'bottom-up' visual saliency model. It is important to use longer videos, when taking into account top-down processes. Moreover, study of audio saliency also requires long videos, as audio modality is processed with a slight delay compared to the visual one.

- The main objective in this thesis was to propose a visual saliency model with improved predictability of eye movements. A perspective is to find practical implications of the model. As an initial step, we develop an efficient implementation of the model using raw computational power of graphics processors. The efficient implementation can then be used for many applications: to develop mobile robots with scene recognition, to assist visually impaired or disabled, to create interest sensitive media and its quality assessment, to facilitate multiple tasks in heads-up displays for IVTs and aerospace augmented with salient objects detection and tracking. Furthermore, a faster implementation will help integrate and test other potential visual features into the existing model—to improve its predictability.

- Next generation of graphics cards by Nvidia, Fermi architecture and the very recent Kepler architecture, extends the computational capabilities of the hardware. With increased double precision compute performance, they are becoming more popular among the GPGPU community. The new architectures with more shared memory per multiprocessor provide a more flexible cache architecture. A perspective is to port the implemented saliency model on to the rapidly evolving GPU technlogy.

- Portability between hardware from different GPU manufacturers is important. Possible future work would be to port the implementation of the model using OpenCL, to test it on AMD GPUs, and to compare the performance against the current CUDA implementation.

- Heterogeneous computing with a host processor supported by a co-processor is adopted in many high performance computing (HPC) systems. Each suitable to solve particular problems with different requirements. An interesting future work could be use heterogeneous computer architectures and programming.

Appendix A

Video databases

A.1 Video stimuli

The visual saliency model is tested against video databases named SM-I [Mar+09], GS-I [SPG11] and GS-II [SPG12] that are assembled using the approach followed by Carmi and Itti [CI06]. Here, each database is composed of videos with varying content from TV shows, TV news, animated movies, commercials, sport and music in indoor, out-door, day-time and night-time conditions. All the videos are decomposed into small clip snippets of several seconds that are randomly joined together into a set of clips of $\sim 30s$. This random fusion of the clip snippets is interesting to study the influences of the early attention rather than the participant anticipating the transitions among the visual frames. The general information regarding the video databases used is illustrated in Table A.1. The main differences between the two video databases are frame size, content and video quality.

	Participants (m/f)	Total clips	Clip snippets per clip	Clip snippet duration	Total frames	Frame size
SM-I	20/10	20	15	1-3s	15082	720×576
GS-I	20/10	10	6	5-8s	10885	608×272
GS-II	18/18	10	8	6-10s	16402	842×474

Table A.1: General information about the video databases used.

❖ **SM-I video database** 53 short videos (25 fps, 720×576 pixels), 15082 frames viewed by 30 participants ($20m/10f$) with age range $23-40$ and had normal or corrected to normal vision. All the videos are decomposed into 305 small clip snippets each $1-3s$ long. The clip snippets are fused randomly to form 20 clips of $30s$ (30.20 ± 0.61). There was not a particular task or question.
Eye positions were recorded at 500 Hz (20 eye positions per frame for two eyes) using a EyeLink II (SR Research). Participants were positioned with their chin supported on a 21" color monitor (70 Hz) at a viewing distance of $57cm$ ($40° \times 30°$ degrees usable field

of view). A calibration was carried out at every five stimuli and a control drift was done before each stimuli.

Figure A.1: Some sample frames from SM-I video database obtained from different video sources, for example: indoor scenes, outdoor, scenes of day and night.

❖ **GS-I video database** Videos from heterogeneous sources (25 fps, 608×272 pixels), 10885 frames viewed by 30 participants ($20m/10f$) with age range $21 - 31$ and had normal or corrected to normal vision. All the videos are decomposed into 60 small clip snippets each $5 - 8s$ long. The clip snippets are fused randomly to form 10 clips of $43s$ (43.22 ± 0.28). There was not a particular task or question.
Eye positions were recorded at 250 Hz (20 eye positions per frame for two eyes) using a Eyelink II (SR Research). Participants were positioned with their chin supported on a 19" color monitor (60 Hz) at a viewing distance of $57cm$ ($20° \times 10°$ degrees usable field of view). A calibration was carried out at every five stimuli and a control drift was done before each stimuli.

Figure A.2: Some sample frames from GS-I video database obtained from different video sources, for example: indoor scenes, outdoor, scenes of day and night.

❖ **GS-II video database** Videos from heterogeneous sources (25 fps, 842×474 pixels), 16402 frames viewed by 36 participants ($18m/18f$) with age range $20 - 34$ and had normal or corrected to normal vision. All the videos are decomposed into 80 small clip snippets each $6 - 10s$ long. The clip snippets are fused randomly to form 10 clips of $65s$ (65.6 ± 0.24). There was not a particular task or question.
Eye positions were recorded at 250 Hz (10 monocular eye positions per frame) using a

EyeLink II (SR Research). Participants were positioned with their chin supported on a 19" color monitor (60 Hz) at a viewing distance of $57cm$ ($35° \times 20°$ degrees usable field of view). A calibration was carried out at every five clips and a control drift was done before each clip.

Figure A.3: Some sample frames from GS-II video database obtained from different video sources, for example: indoor scenes, outdoor, scenes of day and night.

Appendix B

OpenCV face detection

B.1 What is OpenCV

open source computer vision (OpenCV) is an open source library written in C and C++. originally developed by Intel Corporation and afterward taken over by Willow Garage. It is platform independent, and available to be used in commercial products under BSD license. It supports the representation, input and output of images and videos, as well as , set of functions for computer vision and image processing. The main objective of developing the library is to provide real-time computer vision.

B.1.1 Module Summary

The functions available in the latest version of OpenCV 2.4 are divided into different areas:

- **core:** with all the data structures and matrix operations.
- **improc:** with different image manipulation functions.
- **highgui:** with high-level functions to build user interfaces and to read/write images and videos.
- **calib3d:** with functions to extract information about 3D world from 2D information; functions for camera calibration, pose estimation and stereo.
- **video:** for video analysis, such as algorithms for motion estimation and tracking.
- **features2d:** includes framework for feature detection, and descriptor extraction and matching.
- **ml:** with machine learning algorithms for statistical classification, regression and clustering of data.
- **flann:** with algorithms for search and clustering in multi-dimensional spaces.
- **objdetect:** for object detection, for example face and people detectors.

- **gpu:** comprise ports of all functions to take advantage of the computing power of GPUs using CUDA.
- **ocl:** comprise ports of all functions to take advantage of the computing power of GPUs using OpenCL.
- **contrib:** contains newly tested algorithms and contributions made from the community.

B.1.2 Library Characteristics

Some of the main characteristics that make OpenCV interesting for academic and industrial projects are:

- **OpenCV data structures:** Most of the implemented algorithms heavily rely on matrix operations. Hence, for simpler code development and maintenance, the library defines specialized data structures and classes for points, vectors, matrices, and their related operators. Most of these reside in the core module, and are used throughout the library source code.
- **Use of C++ templates:** OpenCV supports matrices of different data types as well as image containers of many image types each with different color channels and depths. The library extensively uses C++ templates to provide an easy to use interface for the functions. It also sets up a flexible platform to extend the existing functionality.
- **Use of C++ standard template library (STL):** Source code of the library is highly simplified and optimized by the extensive use of built-in data structures and functions from STL.
- **Use of streaming SIMD extensions (SSE):** Since most of the matrix operations are data-parallel, OpenCV is heavily optimized and takes advantage of SSE for performance increase. Recently, a couple of modules with algorithms ported onto GPUs to tremendously improve their performance.
- **Ad hoc implementations:** OpenCV is open source, and many contributors from both academic and industry. The core provides all the important data structures, while the already implemented function act as a standard template for development. These combined with many reusable functions are the building blocks to invent more complex algorithms.

B.2 Viola-Jones face detector

B.2.1 Feature definition and extraction

The proposed face pathway uses the Viola-Jones object detector [VJ04]. It is based on the detection of specific features that carry information about the class of object to be detected such as faces, cars, or any other object. The information can be coded by Haar-like features that are sensitive to the orientation of contrasts among regions. In our case, a human face can be represented as a set of features exhibiting the relationship of contrast of different regions like eyes, nose, mouth, etc. The Viola and Jones Haar-like feature set defines 2-rectangle, 3-rectangle and 4-rectangle features. Each feature determines the presence or absence of certain characteristics in the image, such as edges or changes in texture. For example, a 2-rectangle feature can indicate the boundary between a dark region and a light region.

B.2.2 Classifier

A Haar-like feature considers adjacent rectangular regions in the detection window and computes an average pixel value for each region. Then the difference between these values is compared to an already learned threshold to separate non-objects from objects in the detection window. A large number of Haar-like features are required to detect an object robustly, as each represents a *'weak classifier'* or *'low-feature detector'*. Therefore, these *'weak classifiers'* are organized in a cascade of classifiers, which achieves increased detection performance while reducing computation time. Here, the initial cascade starts off with very simple features rejecting the vast majority of image regions. This makes the process simpler, and the cascade becomes more meaningful as it progresses down.

B.2.3 Integral image

Another advantage of using Haar-like features is the use of integral images also known as summed area tables. The advantage of these tables is that the sum and mean of pixel values of an area of arbitrary size can be computed in constant time. Here, each pixel is a sum of the pixels to its upper left region. It can be computed faster, and it is an effective way of calculating the sum of pixel values for the rectangular feature model. For example, the sum of rectangular areas can be computed in the image, at any position or scale, using only four lookups. Likewise, the Viola-Jones 2-rectangle features need six lookups, 3-rectangle features need eight lookups, and 4-rectangle features need nine lookups.

Appendix C

Spatio-temporal fusion

Visual attention is a process to attend to regions in a visual scene that appears to be salient from their surroundings. The map to represent this spotlight of focus in the field of computer vision is called visual saliency map. In the human vision system, the raw information from the visual stimuli is decomposed into several paths that process this information for certain features. At the end of all the processing, these feature maps are combined together into a final visual saliency map that represents the regions of attention. It is important to understand this function of the human vision system, and to create models for computing that can be used to extract relevant information. This capability is potentially applicable in the domains of video compression, video synthesis and analysis, robotics, and many more.

The objective of the work is to have a better understanding of the potentialities of different fusion methods, and to make the best choice for our application. Furthermore, the fusion method must be adaptive to the changing environment and quick to process.

The rest of the work is organized as follows: Section C.1 presents the methods used for information fusion for the two-pathway visual saliency model. Section C.2 presents the findings on the videos, and demonstrates the performance of the different fusion methods evaluated. Section C.3 concludes the work.

C.1 Different fusion methods

The work evaluates six recent fusion techniques for the fusion of static and dynamic saliency maps from the spatio-temporal model. All these intermediate maps have unequal influences in the final visual saliency map, due to the varying input for the separate pathways. Therefore, the motivation behind the evaluation is to find an efficient and robust fusion method that not only extracts all useful information, but also reduces the effects of false findings.

C.1.1 Fusion using Shannon's information theory [HZ07]

Using the Shannon information theory, the conspicuous spots are taken as events. Hence, the information conveyed by each event is calculated by counting the values above a threshold.

This probability is used to yield the information conveyed by each conspicuity map.

$$P(M) = \frac{M > \tau}{M}$$
$$\tau = 0.6 \cdot MAX(M_s \cup M_d)$$

The weights for the static and dynamic map are obtained using:

$$I(M) = -log(P(M))$$
$$W(M) = I(M)MAX(M)$$

and, we get the final map using equation:

$$M_{sd} = W(M_s)I(M_s)M_s + W(M_d)I(M_d)M_d$$

C.1.2 Motion priority fusion model [JQ10]

The work uses the notion of motion priority, as the human vision system pays more attention to the regions in motion against the static background. Here, strong motion contrast will increase the weight for the dynamic map, whereas the fusion weight of the spatial information causes it to decrease. The dynamic weights for the two pathways are calculated as:

$$W_d = \alpha \; exp(1 - \alpha)$$
$$W_s = 1 - W_d$$
$$\alpha = MAX(M_d) - MEAN(M_d)$$

and, then the final saliency map is computed using:

$$M_{sd} = W_s M_s + W_d M_d$$

C.1.3 Binary threshold mask fusion model [Lu+10]

The fusion method uses a mask for the dynamic map, which enhances the robustness when the motion parameters are not estimated correctly. It is useful to exclude the inconsistent regions, and requires no selection of a weighting factor for the spatial and temporal information. Furthermore, the use of MAX operator avoids the suppression of insignificant salient regions.

$$M_{sd} = MAX(M_s, M_d \cap M_{st})$$

Here, $M_{st}(\tau = \bar{M_s})$ is the thresholded static saliency map.

C.1.4 Max skewness fusion model [Mar+09]

The fusion model modulates the static and dynamic saliency maps using the maximum and skewness respectively using:

$$M_{sd} = \alpha M_s + \beta M_d + \gamma M_s M_d$$
$$where, \begin{cases} \alpha = MAX(M_s) \\ \beta = SKEWNESS(M_d) \\ \gamma = \alpha\beta \end{cases}$$

C.1.5 Key memory fusion model [QXG09]

The fusion model uses temporal changes to improve the mean μ and variance S that are calculated as:

$$\mu_s^k = (1-\alpha)\mu_s^{k-1} + \alpha\mu_s^k$$
$$\mu_d^k = (1-\alpha)\mu_d^{k-1} + \alpha\mu_d^k$$
$$S_s^k = (1-\alpha)S_s^{k-1} + \alpha S_s^k$$
$$S_d^k = (1-\alpha)S_d^{k-1} + \alpha S_d^k$$
$$\alpha = \begin{cases} 1/k & 1 \le k \le K \\ 1/K & k > K \end{cases}$$

where K depicts the rate of illumination changes that is set to 2. Whereas, the weight is calculated as:

$$W_k = \frac{(\mu_s^k - \mu_d^k)}{(\delta_s^k + \delta_d^k)}$$

Finally, the fused saliency map is computed as:

$$M_{sd} = W_k M_s + M_d$$

C.1.6 Dynamic weight fusion model [XXR10]

The fusion method uses a dynamic weight calculated from the ratio of the means of static and dynamic maps ($\bar{M}_s$ and $\bar{M}_d$) from the model.

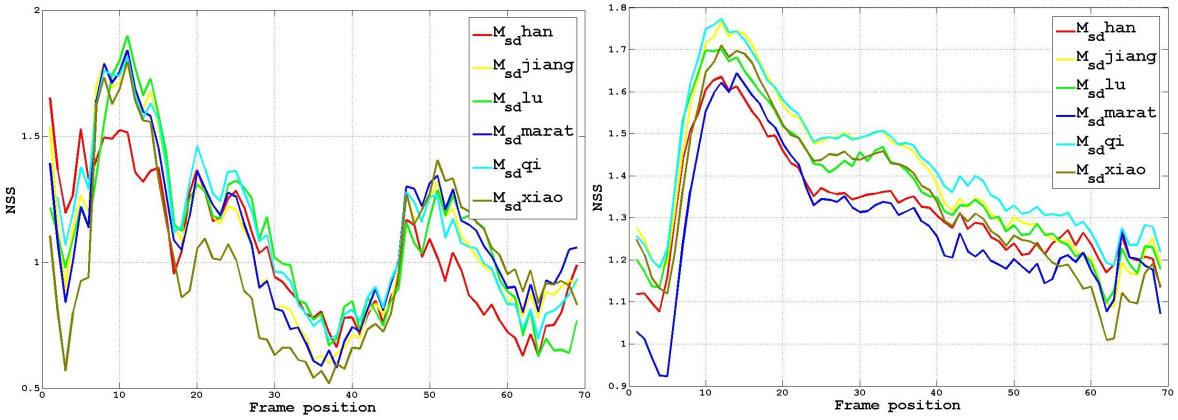
$$M_{sd} = \alpha M_d + (1-\alpha)M_s$$
$$\alpha = \frac{\bar{M}_d}{\bar{M}_s + \bar{M}_d}$$

C.2 Results

Video database	Criterion	M_s	M_d	$M_{sd}han$ [HZ07]	$M_{sd}jiang$ [JQ10]	$M_{sd}lu$ [Lu+10]	$M_{sd}marat$ [Mar+09]	$M_{sd}qi$ [QXG09]	$M_{sd}xiao$ [XXR10]
GS-I	NSS	0.57	1.02	1.02	1.26	1.14	1.19	1.17	1.25
	Gain	-	-	0%	23%	12%	17%	15%	22%
SM-I	NSS	0.88	1.19	1.33	1.40	1.37	1.28	1.43	1.35
	Gain	-	-	12%	18%	15%	7%	20%	13%

Table C.1: Mean NSS for various fusion methods evaluated against two video databases.

The static and dynamic saliency maps are combined into a visual saliency map using different fusion methods described in Section Section C.1. These resulting maps are compared against the experimental eye position density maps using NSS as the criteria.



(a) Evolution of NSS for GS-I video database using various fusion techniques.

(b) Evolution of NSS for SM-I video database using various fusion techniques.

Figure C.1: One face at peripheral (P) or outside (O) locations.

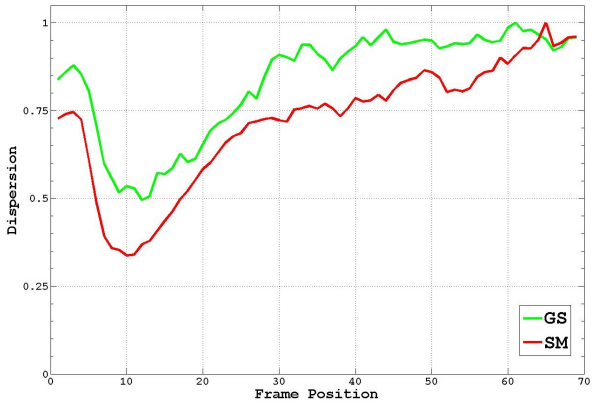
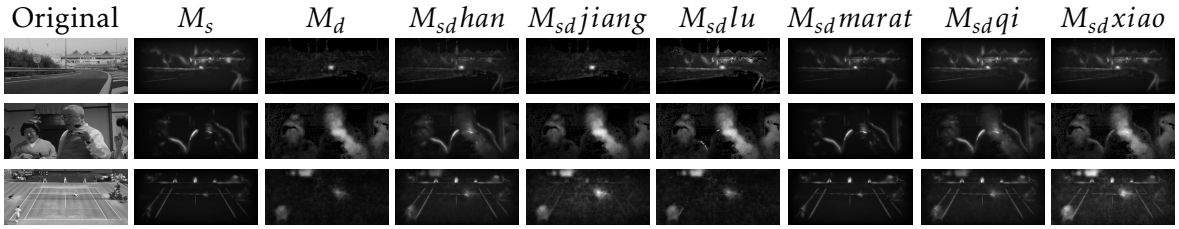


Figure C.2: Dispersion D for eye position as a function of frame for the two video databases.

Here, Figures C.1a and C.1b illustrate the evolution of mean NSS over time for the two video databases, where time is represented by the frame position of the clip snippet (one image = $40ms$). The curve is plotted by averaging the NSS values of the 1st frame of each clip snippet, likewise, the same process is repeated for the first 70 frames of every clip snippet. It is observed that the evolution curve starts off with a low mean NSS value at the beginning of each frame, and it quickly reaches to a peak value at about 13^{th} frame ($520ms$). This phenomenon is explained by the fact that at the change of every clip snippet the real eye positions correspond to the salient regions from the previous clip snippet. The mean NSS curve reaches its maximum value after the involvement of bottom-up influences on the visual stimulus, and then decreases with time. Similarly, Figure C.2 shows that the value of dispersion is high at the beginning of the videos, and it drops to a lowest value as all the participants find a common region of interest. It is significant that about the same time the value of NSS is at a peak value.

Table C.1 shows the NSS mean values and gains after fusion, and Figure C.3 illustrates resulting saliency maps for the two test video databases. Firstly, the fusion methods for $M_{sd}han$ and $M_{sd}lu$ consider a threshold to extract only the useful information from the partial maps. Secondly, we know that in a human visual system attention is often influenced by motion, that is incorporated in the fusion method used for $M_{sd}jiang$. Likewise, $M_{sd}marat$



(a) Examples of saliency maps using different fusion methods for GS-I video database

(b) Examples of saliency maps using different fusion methods for SM-I video database

Figure C.3: Some results for the two video databases

uses skewness as a the motion priority parameter. Thirdly, in $M_{sd}qi$ the fusion used is additive, but the weight from the approximation determines the validity of the maps. Lastly, $M_{sd}xiao$ uses dynamic weight computed as the ratio of the means of static and dynamic maps.

In Figure C.3, the first row for each database represent a scene with high motion, whereas the other samples represent indoor, outdoor and sport scenes. The resulting visual saliency maps show that $M_{sd}jiang$ and $M_{sd}xiao$ give priority to the salient objects in the dynamic saliency map M_d. In the case of indoor and sport scenes, the contours from the static maps M_s and motion from the dynamic maps M_d are fused into final saliency maps depending on the fusion method used.

In Table C.1, the NSS values for partial maps M_s and M_d from the two separate pathways of the model show that globally results for SM-I video database are better than the GS-I video database. Besides this, the results show that the difference of the amount of salient information in the partial maps contributes unequally in the final visual saliency map. This unequal influence is achieved by computing dynamic weights for the partial maps. Resultantly, we get better results for fused maps $M_{sd}jiang$ and $M_{sd}xiao$ for GS-I video database. Whereas, in case of SM-I video database, the fusion maps $M_{sd}qi$ takes into account the quality of both static and dynamic maps. Additionally, the use of a memory effect further enforces the fusion results, and hence we obtain a gain of 20%. Furthermore, we know that dynamic information is important in human visual system, and hence a priority will improve the results. This is observed for GS-I video database, where the fusion maps $M_{sd}jiang$ has the best results.

C.3 Conclusion

The study evaluates six fusion methods for a spatio-temporal model against two test video databases with varying features. In a nutshell, each of the fusion methods has their separate advantages, which could be chosen in function of the application or combined intelligently to result in a method that is more robust.

Appendix D

GPU development

There was a halt in processing capabilities of single machines in the previous decade, consequently rendering unusable to tackle modern day complex tasks. Mainly caused by the physical constraints involved, challenging the predictions of Moore's law. The law predicts that the trend of doubling of computational power can only be achieved by multiplying the number of computing units.

The current transistor design pose certain size limitations, thus an increase in computing power can only be achieved by increasing the number of transistors. Consequently, the predicted trend of exponential growth has decreased. The size and power constraints of the crammed-up transistors on a single chip can be reduced using different techniques. For instance, by adding different caches, or to use instruction-level parallelism. They both provide some performance gains, but the developmental costs of these solutions is higher compared that of the gains achieved.

D.1 Graphics processors

In addition to the hardware parallelization, accelerators are used for specialized tasks. These pieces of hardware offered extreme performance, but they are relatively simpler in design compared to CPU. The most popular of accelerators are GPU—extensively used for graphics processing.

Traditionally, GPUs are use for 2D image rendering and other geometric shapes, where each element is considered standalone data element processed by an individual processing unit of the graphics hardware. This offers lucrative performance gains for embarrassingly parallel computations, which makes it worth consideration for general-purpose computation on graphics processing units (GPGPU).

D.1.1 Specialized processing

In such heterogeneous execution model, the high compute tasks are executed by the co-processors, whereas the rest are executed by the host processors, specifically designed to execute sequential programs. The accelerator cores are used to perform compute-intensive

tasks, very simple and highly parallel. The rest of the complex tasks are handled by the main cores. Therefore, the design of the accelerators is suitable for simplified functionality, as the cores within run at lower clock speed, and they execute computations requiring less resources; consequently consuming less power.

D.1.2 Co-processor

GPUs act as co-processors, or sometimes referred to as accelerators. They are designed to reduce the workload of host processors, as they can execute computations and access memory asynchronously. Thus, it is a heterogeneous system. Over the year, these specialized co-processors or accelerators have become quite popular for interactive simulations, video gaming and 3D rendering platforms. They have evolved from specialized piece of hardware to somewhat generalized computational platform. Additionally, the availability of several development models and their facilitating tools have made GPUs a viable solution GPGPU problems.

D.2 GPU computing

There are two major vendors, NVIDIA and ATi of graphics hardware with their supporting proprietary development toolkits. Recently, an open standard OpenCL is developed to make development easier and portable, irrespective of the underlying graphics processor used.

Here, we present the processing and memory resources for NVIDIA GeForce 285 GTX, the graphics processor used for our GPU implementation. In newer generations of GPUs, the fundamental resources remain similar with increased capabilities. Figure D.1 illustrates a block diagrams of NVIDIA GeForce 285 GTX.

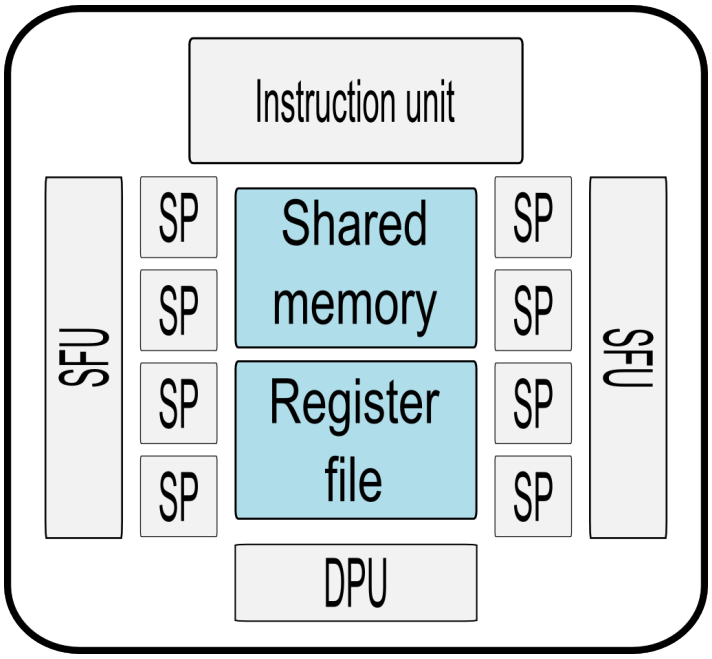


Figure D.1: Block diagram of NVIDIA GeForce 285 GTX graphics processing unit.

D.2.1 Processing resources

- **SP** is a fully pipelined arithmetic logic unit capable of integer and float precision computations.
- **SM** is dual-issue processor with SPs and SFUs working independently. They can handle multiple active warps, and they pose no context change overhead between them. This capability hides the instruction and memory latencies. Furthermore, there are multiple active thread blocks per SM. The number of active blocks is determined by the amount of resources required (shared memory and registers) divided by the total available resources.

  SMs are grouped sharing eight texture units and single L1 cache. Such groups are referred to as texture processing units. Furthermore, there are SFUs with four ALUs capable of vertex attributes interpolation and computing other special transcendentals.

D.2.2 Memory hierarchy

- **Device registers:** The stream processors have a set of local registers. The automatic variables declared inside kernels reside inside these registers, but in some cases the compiler may place these into local memory. This happens when there are either too many register variables that spill out or a structure or array is used that consumes more than the number of available registers, and if an array is not indexed by constants. The access time is zero clock cycles, which is fast.
- **Shared memory space:** Threads within thread block communicate through a local memory, the shared memory. It is organized into 16 banks that can be accessed every clock cycle. Each warp of 32 threads takes four clock cycles to complete. Here, two clock cycles are used up for two shared memory access request—one request per half warp. To avoid conflicts, each thread must access exclusively a memory bank. The accesses are synchronized automatically using barriers.

  The shared memory is on-chip low latency memory. It is accessible to all the SPEs in the multi-processor; providing high performance and communication among the threads of a thread block. Such memory can be implemented effectively in hardware translating to faster memory accesses.

  Shared memory is similar to a local scratch-pad that is 16 KB per block with $16 \times 1KB$ banks, and the threads must accesses must be optimized to avoid bank conflicts. If all the threads of the half-wrap access different banks or the same address, then there is no bank conflict occurring; the memory access times are similar to that of register accesses. Whereas, bank conflicts occur when multiple threads of the same half-warp access the same bank.

  In situations when access to global memory is done multiple times that costs hundreds of clock cycles for each access; it's better to have the values into the shared memory. Also, used to avoid non-coalesced memory accesses by re-ordering them in shared memory.
- **Device memory:**
  - **Global memory** is high latency and optimized memory space for linear accesses. The accesses are preferred to be coalesced; that is, all threads in half warp accessing same 128-bit global memory segment, otherwise memory bandwidth is wasted.

The device or global memory is a large DRAM hard-wired on the graphics hardware; having very high memory bandwidth. The input data is transferred onto this device memory before the kernel execution, and is visible to all the threads invoked by the kernel. Its size varies from device to device.

The memory accesses to global memory takes about 400 to 600 clock cycles, therefore, must be optimized to efficiently utilize the memory bandwidth. It is achieved only when the memory accesses are coalesced, that is, the hardware doing them in minimal transactions. The compute compatibility devices 1.0 and 1.1 can fetch data in single 64-byte and 128-byte transactions respectively. The memory access is coalesced when:

* using 32 | 64 | 128-bit data types
* the starting address and memory alignment i.e. all 16 word transaction lies in the same segment of size equal to the memory transaction size
* the locations accessed by the threads simultaneously are contiguous; referring to the same global memory page

In the case of uncoalesced memory accesses separate transaction is issued for each thread in the half-wrap, causing inevitable performance drops varying with the data type used. An estimated performance drop is 10× for 32-bit, 4× for 64-bit, and 2× for 128-bit data types.

– **Local memory** is a where arrays are placed. It can be accessed dynamically. It incurs large latencies, since it is located in global memory. It is a portion of memory allocated in the global memory by the compiler. It is slower as the memory access is same as the that of global memory region. It resides the thread private or local variables that can't fit within the registers. Its use leads to degradation of performance, which can be discovered by inspecting the PTX assembly code.

– **Texture memory** in GT200 consists of eight 64-bit memory controllers, collectively gives 512-bit memory interface to global memory. Accesses can be directly made to the memory, or through texture cache units as texture cache. Texture comes from traditional GPU design, which were optimized for 2D cache accesses.

Texture memory is read only 8 KB cache space per multi-processor that resides on the global memory. In graphics processing, the designers introduced a specialized hardware to accomplish certain repetitive operation—the texture units. This new piece of hardware performed these operations really fast and it can consider as an extra interface built on top of the global memory. These textures don't impose any restrictions on the access patterns for example memory coalescing; required by other memory spaces. Its efficiency solely depends on 1D/2D locality.

Texture memory gives linear, bi-linear and tri-linear interpolation using dedicated hardware separate from the thread processors. It also converts the integer input data into 32-bit floating points within ranges $[0, 1]$ or $[-1, 1]$, and support configurable return value at the edges using the texture units.

Texture memory space supports integer and floating-point data types. With textures one can exceed the peak performance of global memory, as it accesses the global memory only when a cache miss occurs. It can be useful in situations when the accesses incur a penalty for nearly coalesced accesses i.e. when the starting address is misaligned.

– **Constant memory** is read only 8 KB cached memory space per SM; highly optimized for access to the same location by all the threads. Initially, the data is

written onto the constant cache by the host CPU, and remains persistent throughout the kernel execution; is read-only. It originally resides on the global memory, and visible to all the multiprocessors. Up to 64 KB of data can be placed in constant memory; limitation by the programming model.

The memory space is usually used for lookup tables, where they are cached for efficient access. It is single ported; giving faster access if all threads access the same location, otherwise results in delay. This latency can vary from a single clock cycle to hundreds depending on the locality of the data in the constant cache.

- **Host memory** allows zero-copy accesses to device memory, completely bypassing the CPU.

D.2.3 Programming model

- **SIMT** The single program, multiple data (SPMD) programming model exposed by GPUs combine many individual threads into thread blocks. A simple piece of code, called the kernel, is executed by these blocks. Intra-block communication and synchronization is done through a local memory, the shared memory. On the other hand, inter-block communication is only possible through the global memory.

  The SMs are based on single instruction, multiple thread (SIMT) execution model that allows to write thread-parallel code for independent scalar threads, and data-parallel code for the coordinated threads. It is SIMT that specifies the execution and branching of a single thread. The main idea of SIMT logic is to leave more space for ALUs, consequently, leading to huge performance.

  The host CPU invokes the device kernel as a grid of thread blocks. A compute work distribution (CWD) unit enumerates the blocks, and starts distributing them onto the SMs according to their available execution capacity where they are executed concurrently. Another block is assigned to the vacated multiprocessor when its thread block terminates.

- **Device kernels** The code is composed of host (CPU) and kernel (GPU) code. The host code is responsible for transferring the data to and from the GPU's global memory and afterwards initiates the kernel code through a function call. The kernel code is compiled by the NVIDIA CUDA compiler (NVCC). The structure of parallel code for every single thread is clear and flexible. On the whole, exploits fine-grain data and thread parallelism across the threads nested within coarse-grain data and task parallelism across the thread blocks. This granularity makes the compiled cuda code scalable; executable on any number of processors.

- **Thread Hierarchy** The thread execution model is found to be very effective, as concurrent independent threads express thread parallelism, independent thread blocks show coarse-grain data parallelism, and grids show coarse-grain task parallelism. Threads and thread blocks are declared before invoking a device kernel, thus, giving independence to the parallel code with minimal scheduler overhead.

  - **Grid** is a group of thread blocks, which in turn are groups of threads.
  - **Thread blocks** Each thread block is executed in SIMD fashion on a single SM. The dispatched instruction from the kernel is executed for all the threads in the block across many SPs.

The number of thread blocks executing simultaneously on a multiprocessor are variable, and depends on the available local resources that are distributed among all the threads: for example the number of registers used per thread, and the amount of shared memory required per block. This approach gives a huge number of possible combinations for configurations and optimizations; sometimes becomes a tedious task.

- **Threads** Each thread has a unique local index in its thread block that in conjunction with the index of the thread block determines the array subscripts to work upon.
- **Thread warps** consists of *768 threads* in total. Each SM can manage a pool of *24 wraps*. Active wraps are allotted time-slots by the scheduler, whereas its order of execution is undefined. This time-sliced approach is used to maximize overall utilization of computational resources.

  SIMT unit selects a wrap and broadcasts an instruction from the instruction store to all its active threads. If all the 32 threads in a wrap follow the same execution path then full efficiency is realized. Otherwise, it leads to branch divergence within the wraps, where all the divergent paths are executed serially. Consequently, resulting in unexpected performance drops.
- **Zero-overhead scheduling** is the capability of SM to interleave wraps on instruction-by-instruction basis to hide the memory latencies, and long latency arithmetic operations. If a wraps stalls, SM gets another ready wrap in another block and it stalls only when there is no ready wrap available.
- **Thread divergence** is handled by the hardware using thread masking and serialization. Consequently, this results in lowered resource utilization. Moreover, the impact of divergence is only intra warps, not inter warps.

D.2.4 NVIDIA Kepler

The latest generation of GPUs, Kepler GK110 [Kep] , gives 3.95 teraflops single precision and 1.31 teraflops double precision peak computing performance with fast data transfers at 250 GB/s. There are 15 streaming multiprocessor (SMX) units and six 64-bit memory controllers. Each SMX unit comprise 192 CUDA cores (in total 2688 cores on all 15 SMXs), 32 special function units (SFUs), 32 load/store units (LD/ST), 48 KB of shared memory, 48 KB of constant memory cache, and 255 32-bit registers per thread.

D.3 Why GPUs

Strengths

- **Performance** The main advantage of using GPUs is the computational power at very low cost, compared to other accelerator platforms
- **Inexpensive** A cluster of GPUs is a inexpensive and convenient way to get high-performance computing at with very less special space, cost, and power requirements.
- **Ubiquitous** Being a commodity device, they offer an opportunity for high-performance enthusiasts to learn and experiment.

Weaknesses

- **Memory bottleneck** There is a memory bottleneck for transfers from host memory to device memory. The peripheral component interconnect (PCI) express peak bandwidth performance is considerably low compared to that of the computing power. Also, it is important to hide the latencies of device memories by using most of the device registers, and minimize the memory accesses using caches. Furthermore, to write code with coalesced global memory accesses.
- **Precision** Only recent GPUs support both single and double precision. The use of higher precision, necessary for some scientific problems, may not result in gains expected.
- **Learning curve** Even after the development of more tools facilitating programming for GPU, there is learning curve required. Since the design is changing rapidly every generation, a programmer is required to alter their code according to the memories available and other technological advances added.

Appendix E

Implementation and Usage

E.1 Program options

Synopsis

```
 ./saliency [ --help ] [ --version ] [ --verbose ] [ --one-path-static ]
[ --one-path-dynamic ] [ --one-path-face ] [ --two-path ] [ --three-path ]
[ --save-videos ] [ --save-images ] [ --save-text-files ]
[ --display-input ] [ --display-static ] [ --display-dynamic ]
[ --display-face ] [ --display-master ] [ --fixed-camera ] [ --use-camera ]
[ -i image or video path ] [ -s static image or video path ]
[ -d dynamic image or video path ] [ -f face image or video path ]
[ -m master image or video path ] [ --iext input image or video extension ]
[ --oext output image or video extension ] [ -I n ] [ -N n ]
[ --orientations n ] [ --frequencies n ] [ -p n ] [ -t n ]
[ --cascade-name haar cascade file path ] [ -c configuration file path ]
[ -C camera motion compensation file path ] name
```

Description

The program provides a method to compute visual saliency of a visual scene. It comprises of three separate pathways, each dedicated for a specific visual cue. The pathways are: static, dynamic and face pathway. All the specific saliency information from different pathways is combined or fused together to get the final or master saliency map. This saliency map shows the eye positions prediction made by the attention model.

Input video sequence options (input)

- `-i [ --input-path ] path`
  Image or video path for input images or video frames.
  Warning: only .avi and .mpg videos are supported.

- `-I n`
  Specify the number of the first frame in video sequence.
  Warning: the first frame of a video stream has the number zero.
- `-N n`
  Specify the number of iterations or frames to process in video sequence.
  Warning: the first frame of a video stream has the number zero.
- `--iext path`
  Image or video extension for input images or video frames.
  Warning: only .avi and .mpg videos are supported.
- `--oext path`
  Image or video extension for output images or video frames.
  Warning: only .avi and .mpg videos are supported.

Motion model options (input)

- `--one-path-static`
  Static pathway of visual saliency model.
- `--one-path-dynamic`
  Dynamic pathway of visual saliency model.
- `--one-path-face`
  Face pathway of visual saliency model.
- `--two-path`
  Two pathway visual saliency model.
- `--three-path`
  Three pathway visual saliency model.

Warning: only one model option can be used at any time.

Static pathway options (input)

- `--frequencies n`
  Number of frequencies.
  Warning: not implemented.
- `--orientations n`
  Number of orientations.
  Warning: not implemented.

Dynamic pathway options (input)

- `-p [ --pyramid-levels ] n`
  Number of pyramid levels.
  Warning: not implemented.
- `-t [ --temporal-len ] n`
  Size of temporal median filter.

- `--fixed-camera`
  Deactivate camera motion compensation.

Face pathway options (input)

- `--cascade-name path`
  Haar cascade file path.

Result options (output)

- `-s [ --static-path ] path`
  Image or video path for static saliency maps.
- `-d [ --dynamic-path ] path`
  Image or video path for dynamic saliency maps.
- `-f [ --face-path ] path`
  Image or video path for face saliency maps.
- `-m [ --master-path ] path`
  Image or video path for master saliency maps.
- `--save-images`
  Save saliency maps as images.
- `--save-videos`
  Save saliency maps as videos.
- `--save-text-files`
  Save saliency maps as text files.

Warning: when multiple save options are used, all results for a particular pathway are written into the output path specified.

Other options

- `-v [ --verbose ]`
  Activate the verbose mode.
  Warning: not implemented.
- `--version`
  Print the version string.
- `-h [ --help ]`
  Print the help.
- `--display-input`
  Visualize input frames.
- `--display-static`
  Visualize static saliency maps.
- `--display-dynamic`
  Visualize dynamic saliency maps.

- `--display-face`
  Visualize face saliency maps.
- `--display-master`
  Visualize master saliency maps.
- `--use-camera`
  Video stream from webcam. [-c configuration file path]
- `-c [ --config ] path`
  Default configuration file path.

E.2 Example use

- `./saliency -i /home/videos name`
  Compute saliency maps for video and visualize the fused master saliency.
- `./saliency -use-camera`
  Compute saliency maps for video stream from webcam and visualize the fused master saliency.
- `./saliency --display-static -i /home/videos name`
  Compute saliency maps for video, and visualize the fused static and master saliency maps.
- `./saliency --save-images -i /home/videos -s /home/images/static -m /home/images/master -t 5 name`
  Compute saliency maps for video, visualize the fused master saliency map, and save static and master saliency maps as images.
- `./saliency --save-videos -i /home/videos -s /home/videos/static -m /home/videos/master -t 5 name`
  Compute saliency maps for video, visualize the fused master saliency map, and save static and master saliency maps as video.

Appendix F

Résumé en français

Les études menées dans cette thèse portent sur le rôle des visages dans l'attention visuelle. Nous avons cherché à mieux comprendre l'influence des visages dans les vidéos sur les mouvements oculaires, afin de proposer un modèle de saillance visuelle pour la prédiction de la direction du regard. Pour cela, nous avons analysé l'effet des visages sur les fixations oculaires des observateurs regardant librement (sans consigne ni tâche particulière) des vidéos. Nous avons étudié l'impact du nombre de visages, de leur emplacement et de leur taille. Il est apparu claire que les visages dans une scène dynamique (à l'instar de ce qui se passe sur les images fixes) modifient fortement les mouvements oculaires. En nous appuyant sur ces résultats, nous avons proposé un modèle de saillance visuelle, qui combine des caractéristiques classiques de bas-niveau (orientations et fréquences spatiales, amplitude du mouvement des objets) avec cette caractéristique importante de plus haut-niveau que constitue les visages. Enfin, afin de permettre des traitements plus proches du temps réel, nous avons développé une implémentation parallèle de ce modèle de saillance visuelle sur une plateforme multi-GPU. Le gain en vitesse est d'environ 100 par rapport à une implémentation sur un cœur de processeur multithread.

F.1 Introduction

Les êtres humains ont cinq sens traditionnels. La vision, l'ouïe, l'odorat, le touché; chacun provenant d'un système dédié sensoriel ou d'un organe. Seule une fraction de l'information entrante est choisie par un processus de sélection très développé et sophistiqué appelé «l'attention». Deux facteurs sont considérés pour conduire la sélection: (1) la capacité inhérente des stimuli pour capter l'attention, et (2) l'importance comportementale de l'information à des objectifs ou des tâches à accomplir. En premier lieu, l'objectif de sélection est de consacrer les ressources limitées du cerveau seulement à des informations importantes.

Le système visuel humain est censé être responsable du traitement de 90% des informations utiles pour la perception et, par conséquent, a une importance considérable dans la vie quotidienne. Cependant, notre système visuel a une capacité limitée à traiter toutes les informations captées par la rétine. Pour fonctionner et pour accomplir des tâches, il est important d'extraire une partie pertinente de cette information. Ce type spécifique de la sélection et de focalisation des ressources de traitement du cerveau sur une petite région,

analogue à un projecteur, est appelé «attention sélective visuelle». Le spot de focalisation est plus que le choix d'une région d'intérêt, il soit capturé par les caractéristiques de bas niveau d'une région, ou est attiré vers un objet selon des exigences de la tâche. Il implique à la fois des mécanismes bottom-up et top-down, ainsi que leurs interactions, exécutant une capture attentionnelle en planifiant et en effectuant des mouvements oculaires.

F.2 Problème traité

La notion d'attention sélective ayant une grande importance dans le domaine de la vision par ordinateur, est un domaine de recherche actif. Le but principal de la sélection d'une région d'intérêt est l'omission d'informations non désirées et redondantes, afin d'améliorer la qualité du traitement visuel, tout en consommant moins de ressources. Il s'agit d'une haute collaboration de processus parallèles exogènes, et des mécanismes endogènes. Les mécanismes sont extrêmement efficaces, mais ils sont assez complexes à imiter pour les systèmes artificiels. Par conséquent, des études approfondies afin d'accroître notre compréhension de l'attention visuelle est requise.

Au fil des années, plusieurs modèles ont été proposés simulant les processus pré-attentif ou exogène. La plus part de ces modèles sont pilotés par les caractéristiques de bas niveau comme l'intensité, la couleur, les orientations, le mouvement, etc. Les valeurs de ces caractéristiques déterminent la saillance de chaque région d'une scène visuelle. En dehors de ces caractéristiques, les visages peuvent fournir les informations les plus pertinentes en raison de leur importance sociale et évolutive. Les informations concernant l'âge, le sexe, la race, les émotions, l'attractivité et la direction du regard extraites de l'information faciale peuvent orienter l'attention. Certaines études ont montré que les visages sont traités comme les autres de bas niveau, au cours des premières étapes des transformations rétinales et corticales. De plus, la présence des circuits neuronaux spécialisés pour la détection du visage suggèrent leur capacité élevée de capture attentionnelle, quelles que soient les exigences de la tâche. Nous considérons que tous ces facteurs contribuent à mettre l'accent sur les visages, en allouant davantage de ressources attentionnelles aux visages dans le système visuel.

- Le premier défi, dans cette thèse, est d'étudier l'intérêt des visages à l'aide d'expériences et d'évaluations. Il y a plusieurs questions quant à savoir si la répartition de l'attention sur les visages est une fonction de leurs emplacements, leur nombre ou leur taille. Ils sont sans doute spéciaux, mais quel est l'impact de ces attributs? Comment l'attention est dirigée quand un visage est rencontré? En outre, que peuvent-ils transmettre? Est-ce la saillance des visages dépendante à la tâche ou non? Quelles caractéristiques significatives du signal du visage influencent l'attention? Quel type de visage capte l'attention?

- Le deuxième défi consiste à intégrer les résultats des expériences et des évaluations dans un modèle de calcul, ou «modèle de saillance visuelle». Dans cette thèse, le modèle étudié est bottom-up qui calcule la saillance visuelle d'une scène. Il sort une carte avec toutes les régions saillantes mises en évidence, qui peut être utilisée pour étendre et réduire la complexité des algorithmes de vision artificielle. L'idée ici est de prétraiter les informations visuelles, puis d'appliquer le calcul à un seul morceau important ou à l'information importante.

- Le troisième défi est l'intégration de nouveaux attributs et l'amélioration du modèle d'attention visuelle, ce qui augmentera la complexité des modèles existants. Par

conséquent, il est important que les calculs du modèle soient efficaces, pour que le modèle soit facilement et rapidement utilisé par les chercheurs. L'objectif ici est de calculer rapidement et d'évaluer la performance du modèle en prédiction des mouvements oculaires. Une solution avérée est de proposer une version parallélisée sur processeurs graphiques du modèle, qui prend en compte le facteur de vitesse, la faisabilité et la puissance. Au final, l'objectif est de construire un modèle de saillance visuelle rapide, précis, robuste, maniable et accessible.

Le principal objectif de la thèse est d'étudier l'intérêt des visages, et d'intégrer un canal dédié visage dans un modèle de saillance visuelle. En fin de compte, une mise en œuvre efficace du modèle avec parallélisation sera utile pour les systèmes de vision temps réels.

F.3 Objectif

Compte tenu des défis mentionnés, nous avons clairement identifié le problème dans un sens large, et proposé une solution impliquant différents domaines de la science de la cognition à la vision par ordinateur.

- La première étape consiste à effectuer des expériences de psychophysique de l'homme impliquant des études quantitatives du comportement. Les études identifient les différentes fonctionnalités sous-jacentes traitées par les mécanismes attentionnels dans le cerveau humain. Il aborde également la pertinence des caractéristiques communes identifiées à la perception. Enfin, les résultats de ces études peuvent être utilisés pour construire un modèle de l'attention visuelle.

- La deuxième étape consiste en une modélisation informatique, ainsi qu'à développer une description du comportement observé de l'attention visuelle chez humains. Le modèle qui en résulte est soit évalué par rapport aux mouvements oculaires des sujets, ou par rapport aux autres modèles qualitativement. Dans la généralisation des mécanismes de déploiement de l'attention il est important d'étudier la saillance des différentes caractéristiques de la scène visuelle. Les sorties des modèles de l'attention visuelle sont souvent des cartes de saillance qui identifient les régions saillantes principales. Ces cartes peuvent avoir un impact important sur la vision par ordinateur et industrielle. Par exemple, dans les robots cognitifs, intégrer un modèle artificiel de l'attention humaine permet aux robots de prendre des décisions en temps réel selon les informations cognitives sélectionnées de l'environnant.

- La dernière étape met en œuvre un système de vision qui fonctionne à faible résolution en utilisant l'attention sélective comme une étape de prétraitement Dans les applications comme les robots autonomes et des véhicules, la reconnaissance d'objets, suivi de cible, surveillance de la sécurité et de l'inspection industrielle, il est important que l'étape de sélection soit en temps réel ou relativement rapide. Toutes ces applications bénéficient de la sélection en vue d'obtenir une allocation efficace des ressources en utilisant uniquement les informations pertinentes.

F.4 Les contributions principales

La thèse s'appuie sur les travaux précédents de la modélisation de la saillance visuelle. Une base de données vidéo déjà compilées et les mouvements d'oculomètre correspondants

enregistrés pendant l'observation libre des vidéos sont utilisés. Le modèle proposé est étendu pour augmenter sa prévisibilité. Voici les principaux apports de cette thèse.

- Nous étudions l'influence du visage sur la direction du regard dans les vidéos, particulièrement l'impact d'emplacement et du nombre des visages. Nous avons utilisé les données de mouvements oculaires obtenues d'une expérience oculométrique lors d'observation libre des vidéos. Nous avons également étiqueté manuellement tous les visages dans les vidéos visionnées. Nous avons comparé les positions des yeux et l'emplacement des visages à l'aide de différents critères. Pour l'analyse, nous avons considéré les cadres avec un ou deux visages. Dans les deux cas, les scores diminuent avec une excentricité croissante. Toutefois, dans le cas de deux visages, le score du visage ayant une excentricité inférieure est plus élevé par rapport à son homologue. En outre, nous avons analysé les durées de fixation pour un ou deux visages. Nous constatons que les visages dans les vidéos conduisent à de longues fixations, beaucoup plus longues dans le cas d'un visage par rapport au cas de deux visages. En outre, dans le cas d'un visage les fixations à faibles excentricités durent plus longtemps par rapport au cas des deux visages.

- Les visages jouent un rôle important dans l'orientation de l'attention visuelle, par conséquent, l'inclusion de la détection de visage dans un modèle d'attention visuelle classique peut améliorer la prédiction des mouvements oculaires. Dans cette thèse, nous avons proposé un modèle de saillance visuelle pour prédire les mouvements oculaires pendant l'exploration libre des vidéos. Le modèle est inspiré de la biologie du système visuel. Il décompose chaque frame d'une base de données vidéo en trois cartes de saillance, chacune présentant une fonction particulière visuelle. (a) Une carte de saillance «statique» met l'accent sur les régions qui diffèrent de leur contexte en termes de luminance, d'orientation et de fréquence spatiale. (b) Une carte «dynamique» souligne la saillance du déplacement des régions avec des valeurs proportionnelles à l'amplitude de mouvement. (c) Une carte de saillance «visage» met l'accent sur les régions où un visage est détecté avec une valeur proportionnelle à la confiance de la détection. Ici, les deux premières voies permettant de calculer les cartes de saillance statique et dynamique sont empruntées au modèle de saillance visuelle de Marat [Mar10; Mar+09]. Nous avons également comparé les mouvements oculaires avec les cartes de saillance crées par d'autres modèles pour quantifier leur efficacité. Nous avons aussi examiné l'influence du biais central sur les cartes de saillance et incorporé ce biais dans le modèle d'une manière appropriée. Enfin, nous avons proposé une méthode de fusion efficace de toutes ces cartes de saillance. La carte de saillance finale (master) développée dans cette recherche est un bon indicateur de positions des yeux des participants.

- Le modèle de saillance visuelle proposé est complexe et intense en termes de calcul. il faut donc une implémentation plus rapide afin de pouvoir analyser les résultats rapidement ainsi que progresser efficacement dans l'analyse de performance du modèle. Afin de pouvoir répondre à ces besoins, nous proposons une implémentation très efficace de ce modèle avec multi-GPU. Nous présentons les algorithmes du modèle ainsi que plusieurs optimisations parallèles sur GPU avec une plus grande précision et des résultats accélérés de temps d'exécution. L'exécution en temps réel de ce modèle multi-voies sur un multi-GPU en fait un outil puissant pour faciliter la mise en œuvre d'applications de vision.

F.5 Bilan des différentes études

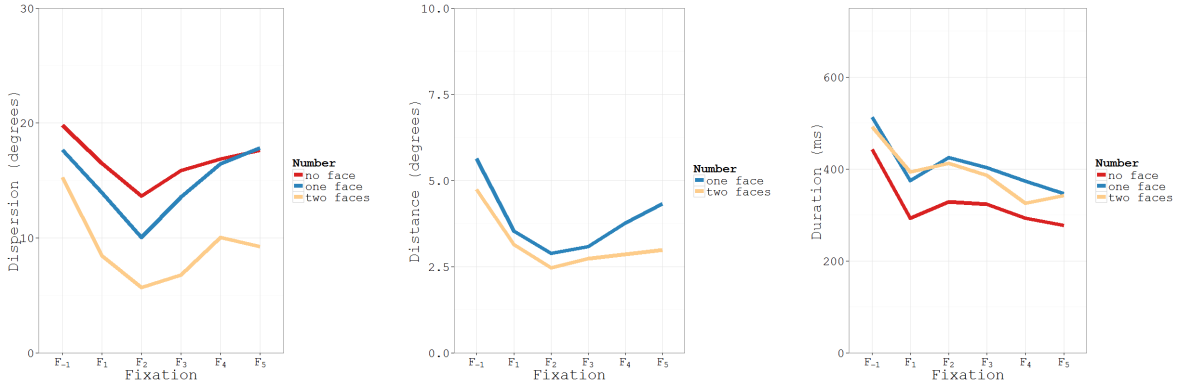
Dans cette thèse, nous avons proposé un modèle de saillance visuelle pour déterminer les régions saillantes des vidéos, basé sur les observations à partir des données des mouvements oculaires. Les données ont été acquises au cours de l'observation libre des vidéos. En conclusion, nous avons des résultats qui ont conduit à plusieurs contributions et leurs perspectives connexes.

Dans le chapitre 3, nous avons évalué la préférence des visages dans les vidéos. L'étude a utilisé des données provenant d'une des expériences psycho-visuelles pour examiner les tendances du regard des participants.

- ❖ Les visages attirent l'attention dans les vidéos. Nous avons constaté que les fixations sur le début de la scène correspondent aux régions d'intérêt dans la scène précédente, entraînant une dispersion de fixation plus élevée chez les participants (Figure F.1a). Comme la scène progresse, la dispersion diminue. Il est beaucoup plus faible pour les scènes avec des visages.

- ❖ Les régions du visage sont saillantes dans les vidéos. L'évaluation en utilisant des critères différents de comparaison montre que les fixations sont faites à proximité des régions du visage (Figure F.1b). Nous concluons que cela est essentiellement lié à l'importance informationnelle et sociale des visages.

- ❖ Les fixations sont plus longues sur les visages dans les vidéos, en particulier sur les scènes avec un seul visage. Nous observons que les fixations initiales sont plus courtes par rapport aux fixations suivantes. Il semble susceptible que les premières fixations soient influencées par la tendance des sujets à chercher le centre de la scène visuelle au début de l'observation, ce qui entraîne des fixations courtes. Les fixations suivantes sont plus longues, car ils sont faits pour extraire un maximum d'informations faciales (Figure F.1c). Cependant, les fixations sont plus courtes quand plusieurs régions d'intérêt ou deux visages sont en concurrence pour les ressources attentionnelles limitées.



(a) Dispersion de fixation entre les participants

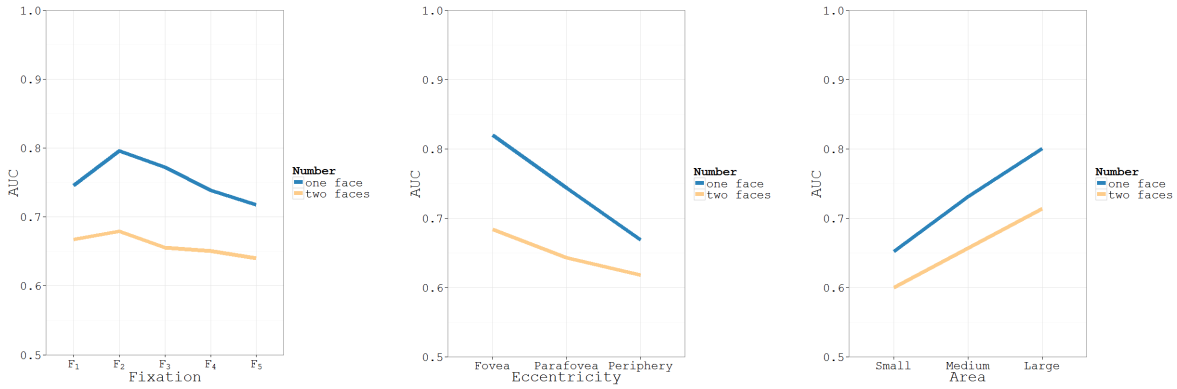
(b) Distance minimum de fixation à visage.

(c) Durée de fixation.

Figure F.1: Mesures d'évaluation pour une et deux visages. Nous avons pris les cinq premières fixations $\{F_1, F_2, F_3, F_4, F_5\}$ après le début de la scène actuelle et la fixation F_{-1} de la scène précédente (fixation juste avant le début de la scène actuelle).

- ❖ Nous déclarons que la préférence des visages dans des scènes dynamiques est influencée par différents facteurs, tels que l'excentricité, la taille et le nombre de visages apparus

dans la scène (Figure F.2). Nous montrons que l'influence des visages diminue avec l'augmentation de l'excentricité. Il est relativement plus faible pour les scènes avec un seul visage par rapport à des scènes avec deux visages. Dans ce dernier cas, le facteur d'excentricité de fixation est utilisé pour résoudre la concurrence entre les deux visages. Nous confirmons que l'augmentation de la superficie des visages améliore leur performance visuelle en masquant les effets de l'excentricité et de la concurrence.



(a) Nombre de visages (b) L'excentricité de visage (c) Surface du visage

Figure F.2: Les scores pour les critères d'évaluation des AUC en fonction de différents facteurs qui influencent pour une ou deux visages.

Dans le chapitre 4, nous utilisons les observations sur l'influence des visages dans les vidéos de proposer une voie de visage.

- Nous proposons un nouveau modèle de saillance visuelle bottom-up qui décompose le signal visuel en utilisant trois voies de traitement en fonction de différents types de caractéristiques visuelles (Figure F.3): statique, dynamique, et visage. Il s'agit d'une extension du modèle de saillance proposé par [Mar+09]. Les voies statiques et dynamiques sont inspirées par la biologie des premières étapes du système visuel humain: la rétine comme un filtre et le cortex comme une banque de filtres. La voie statique extrait les informations de texture basée sur la luminance. La voie dynamique extrait des informations concernant les mouvements des objets. La voie visage extrait des informations sur la présence des visages dans les frames. Le modèle intègre également le biais central comme une modulation adaptée de la carte de saillance visuelle.

- Nous avons évalué la voie du visage proposé par rapport aux données de mouvements oculaires de l'expérience psycho-visuelle. Nous montrons que l'inclusion des visages améliore le modèle de saillance visuelle bottom-up.

- L'expérience oculométrique nous permet d'étudier les caractéristiques visuelles qui attirent le regard d'un participant, et trouver la meilleure façon de les intégrer dans le modèle de saillance. Cela nous permet également de concevoir une fusion efficace et robuste des trois types de cartes en une seule carte de saillance maître. Les coefficients retenus pour les trois voies, statique, dynamique et visage sont le maximum, le skewness et la confiance respectivement. Ces coefficients permettent de renforcer les cartes les plus pertinentes dans la carte de saillance maître (Figure F.4).

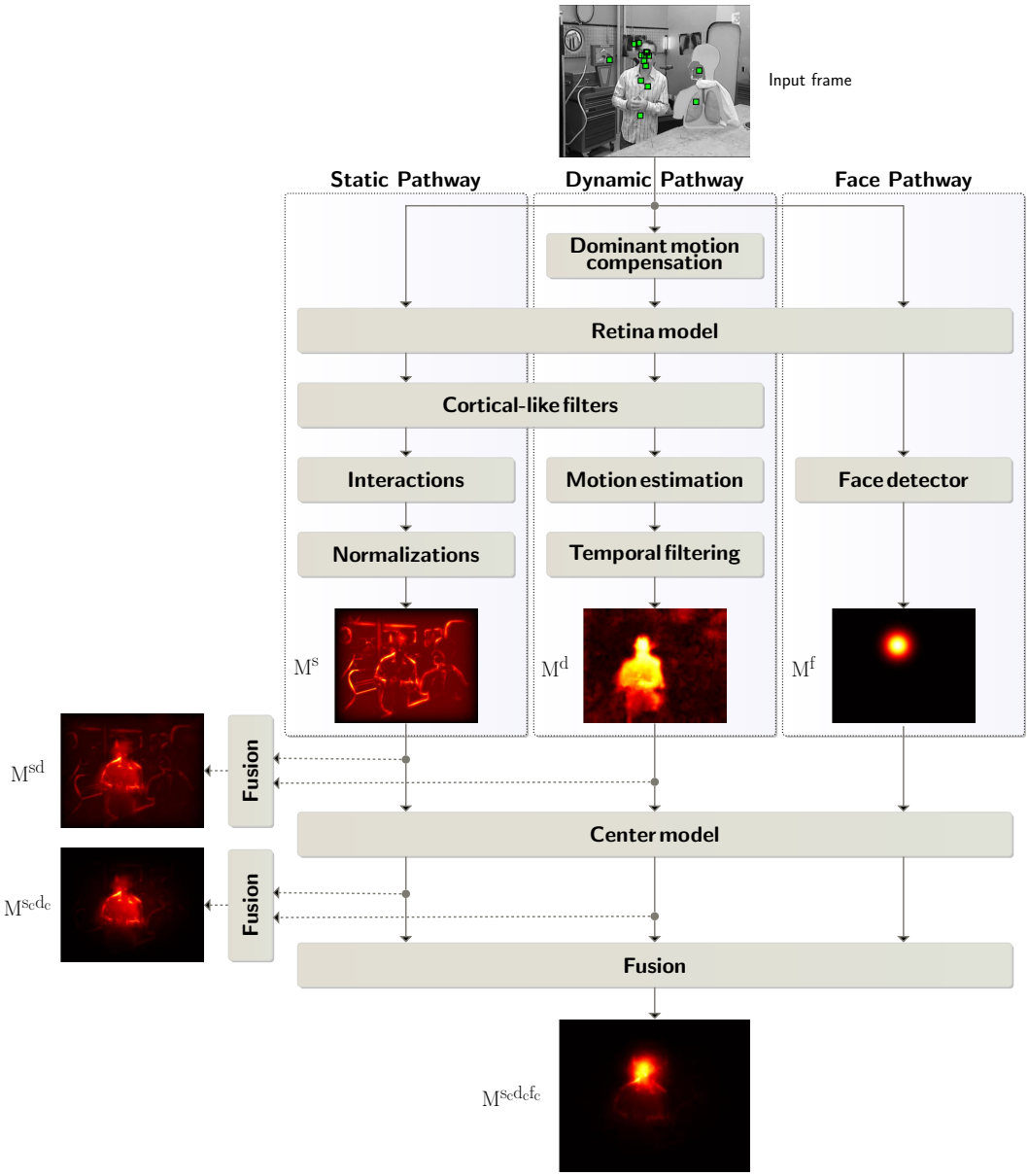


Figure F.3: Schéma bloc du modèle de saillance visuelle proposée avec trois cartes de saillance dédiées à des fonctions spécifiques: statique, dynamique, et le visage.

Dans le chapitre 5, nous présentons une implémentation multi-GPU du modèle de saillance visuelle.

- ❖ Les résultats montrent que le modèle de saillance visuelle à base de GPU proposé surpasse une application à base de CPU équivalente jusqu'à 132×. À notre connaissance, c'est la première application à base de GPU du modèle de saillance jamais rapportée dans la littérature.
- ❖ Les gains de performances sur les GPUs peuvent être obtenus après un examen attentif de configuration des threads et blocs, ainsi que l'allocation efficace d'accès à la mémoire globale et à la mémoire partagée. Les résultats de l'évaluation du code sur GPU optimisé montrent une plus grande efficacité par rapport à la version non optimisée du programme.

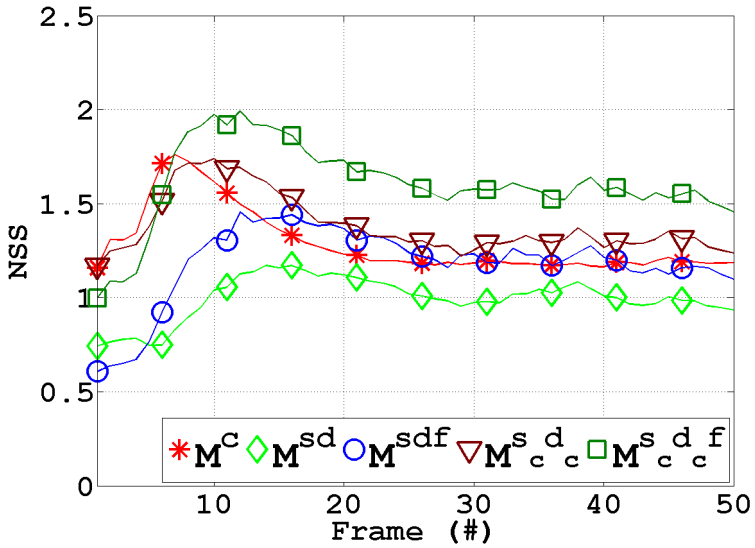


Figure F.4: Evolution des métriques NSS pour des voies différentes, avec ou sans le biais de centre de base de données vidéo.

❖ La mise en œuvre initiale du modèle de saillance visuelle a été réalisée en MATLAB. Cette implémentation a d'abord été portée en C séquentiel, qui obtient une amélioration considérable et aussi permet d'écrire des programmes en CUDA. les noyaux CUDA ont été utilisé pour isoler toutes les parties parallèles du modèle. La mise en œuvre finale sur GPU permet d'augmenter la performance par rapport aux implémentations CPU, comme l'illustre la Figure F.5. Les temps d'exécution pour les différentes implémentations sont résumées dans le Tableau F.1.

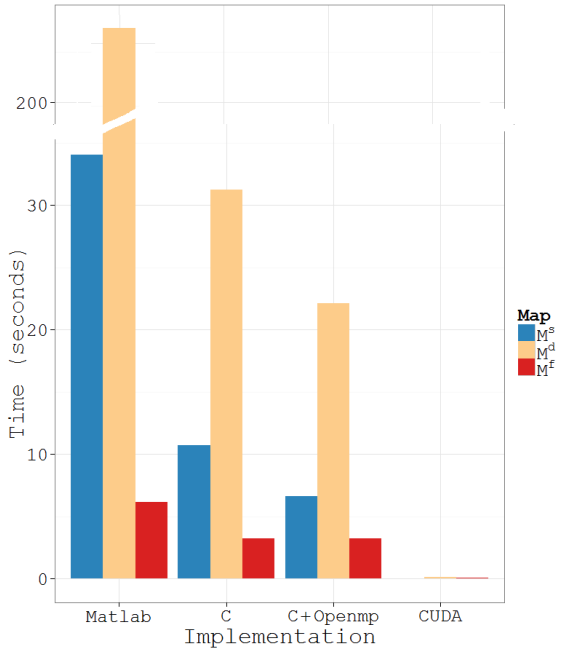


Figure F.5: Timings d'implémentations séquentiels et parallèles pour la vidéo avec la taille d'image 640×480 sur NVIDIA GeForce GTX 285.

	M^s	M^d	M^f
MATLAB	34.01	237.03	6.18
C	10.73	31.24	3.26
C+OpenMP	6.65	22.13	3.21
CUDA	0.04	0.12	0.07

Table F.1: Timings de différentes implémentations séquentiels et parallèles pour la vidéo avec la taille d'image 640×480 sur NVIDIA GeForce GTX 285.

En conclusion, le principal avantage d'amélioration du gain de performances est qu'il permet l'inclusion des autres caractéristiques visuelles dans le modèle et l'évaluation rapide du modèle par rapport aux données expérimentales des positions des yeux. En outre, la solution rapide peut être utilisée dans une grande variété de problèmes de la recherche et de l'industrie.

F.6 Perspectives et travaux futurs

Compte tenu de ces contributions, nous pouvons dire que les objectifs initiaux de cette recherche ont été respectés. Le tableau suivant présente les plans possibles pour les travaux futurs, lié à la présence des visages dans les vidéos et à l'implémentation des modèles de saillance:

- Les caractéristiques de Haar utilisées par le détecteur de visage ne sont pas biologiquement inspirées. Dans cette étude, nous avons utilisé un détecteur de visage de Viola-Jones comme point de départ parce qu'il est populaire, rapide, robuste, et surtout il a une implémentation parallèle disponible. Par conséquent, l'algorithme a été utilisé pour proposer une voie de visage pour le modèle existant avec de bons résultats de détection. La perspective est de proposer une voie de visage à l'aide des algorithmes de reconnaissance des visages biologiquement inspirés, tels que les neurones [VR+98] et les filtres de Gabor [Phi+00].

- Certaines études ont observé que l'utilisation de stimuli multi-modaux, dans les conditions des interactions sociales peut aider à rendre les mouvements oculaires plus précis et plus rapides [Cor+02; AC03; Bel+11; Cou+12; SPG12; Vo+12]. Par exemple, pendant l'observation des scènes avec des visages et du son, les mouvements oculaires sont dirigés vers les régions de la bouche pour faciliter intelligibilité de la parole en utilisant la lecture labiale. En revanche, dans les scènes sans son les yeux sont regardé plus souvent [Jef96; Vo+12]. Par conséquent, l'inclusion de deux modalités audiovisuelles dans un modèle de saillance peut améliorer sa performance. Il est aussi, plus plausible d'utiliser un tel modèle audiovisuel pour les scènes avec du son, qui sont souvent rencontrées dans la vie réelle.

- Un modèle à différentes caractéristiques de bas niveau et de haut niveau nécessite une méthode efficace pour combiner l'information. Différentes méthodes de fusion ont leurs avantages et inconvénients selon les bases de données vidéo avec des attributs différents. Un point de vue consiste à définir une fusion robuste pour toutes les voies du modèle en fonction de l'application du modèle.

- Le modèle proposé est basé sur des caractéristiques visuelles de bas niveau « bottom-up». Sa performance de prédiction peut être augmentée en intégrant différents processus de haut niveau, comme la mémoire de travail (WM) et la mémoire à court terme (STM) du système visuel. La perception de l'objet est un autre exemple de l'information de haut niveau avec des voies dorsales et ventrales qui interagissent pour terminer la détection d'objet et les tâches de reconnaissance. Il sera intéressant d'intégrer certaines de ces informations dans le modèle existant, pour prédire les mouvements oculaires pendant l'observation de longue durée d'une scène avec ou sans tâche.

- Nous avons utilisé des bases de données vidéo comprenant des extraits de vidéo de courte durée pour évaluer le modèle de saillance visuelle proposée «bottom-up». Il est important d'utiliser des vidéos plus longues, si l'on tient compte des processus top-down. Par ailleurs, l'étude de saillance audio nécessite également des vidéos longues, car la modalité audio est traitée avec un léger retard par rapport à la modalité visuelle.

- L'objectif principal de cette thèse est de proposer un modèle de saillance visuelle avec amélioration de la performance de la prédiction des mouvements oculaires. Une perspective est de trouver des applications pratiques du modèle. Comme première étape, nous développons une implémentation efficace du modèle en utilisant la puissance de calcul brute des processeurs graphiques. Cette implémentation peut ensuite être utilisée pour de nombreuses applications: développer des robots mobiles avec reconnaissance de scène, aider les malvoyants ou les handicapés, créer des médias selon l'intérêt des utilisateurs et évaluer leur qualité, faciliter l'exécution des tâches multiples sans intervention de l'utilisateur dans la technologie embarquée à bord des véhicules (in-vehicles technologies en anglais) et aussi dans les technologies aérospatiales en augmentant le mécanisme d'attention avec détection des objets saillants et leur suivi. En outre, une implémentation plus rapide aidera à intégrer et tester d'autres caractéristiques visuelles potentielles dans le modèle existant, afin d'améliorer sa performance de prédiction des mouvements oculaires.

- La prochaine génération de cartes graphiques NVIDIA de l'architecture Fermi et l'architecture Kepler très récente, étend les capacités de calcul du matériel. Avec l'augmentation de la performance de calcul de double précision, ils sont de plus en plus populaires parmi la communauté GPGPU. Les nouvelles architectures à mémoire partagée par multiprocesseur offrent une architecture de cache plus souple. Une perspective est de porter le modèle de saillance sur les nouvelles technologies de GPU qui évoluent rapidement.

- La portabilité entre le matériel des différents fabricants de GPU est importante. Un des travaux futurs possibles serait de porter l'implémentation du modèle sur OpenCL, de le tester sur les GPU AMD, et de comparer la performance contre l'implémentation actuelle CUDA.

Acronyms

ALU	arithmetic logic unit
AMD	Advanced Micro Devices
ANOVA	analysis of variance
API	application programming interface
APP	accelerated parallel processing
AUC	area under the curve
BSD	Berkeley software distribution
CC	cross correlation
Cell	cell broadband engine architecture
CPU	central processing unit
CUDA	compute unified device architecture
cuFFT	NVIDIA CUDA fast Fourier transform library
CWD	compute work distribution
DRAM	dynamic random access memory
FFA	fusiform face area
FFT	fast Fourier transform
FFTW	fastest Fourier transform in the west
FIT	feature integration theory
FPR	false positive rate
GAFFE	gaze-attentive fixation finding engine
GB	gigabyte
GBVS	graph-based visual saliency
GHz	gigahertz
GPGPU	general-purpose computation on graphics processing units
GPU	graphics processing unit
GS-I	Guanghan Song's video database I
GS-II	Guanghan Song's video database II
HVS	human visual system
IOC	inter-observer congruency
IT	inferotemporal cortex
IVT	in-vehicles technologies
KL	Kullback Leibler divergence
LD/ST	load/store units
LGN	lateral geniculate nucleus
MATLAB	matrix laboratory
MCT	mean census transform
MT	middle temporal
NSS	normalized saliency scanpath
NVCC	NVIDIA CUDA compiler
OpenCL	open computing language
OpenCV	open source computer vision
OpenMP	open multi processing
PCI	peripheral component interconnect
PFC	prefrontal cortex
PPC	posterier parietal cortex
PTX	parallel thread execution
RAM	random-access memory
ROC	receiver operating characteristic
SC	superior colliculus
SDK	software development kit
SFU	special function units

GT200	GeForce Tesla 200 series
SIMD	single instruction, multiple data
SIMT	single instruction, multiple thread
SM	streaming multiprocessor
SMX	streaming multiprocessor
SM-I	Sophie Marat's video database
SP	scalar processor
SPE	synergistic processor elements
SPMD	single program, multiple data
SSE	streaming SIMD extensions
STL	standard template library
STM	short-term memory
TC	Torralba's percentile criterion
TPR	true positive rate
V1	primary visual cortex
V2	visual area V2
V3	visual area V3
V4	visual area V4
V5	visual area V5
VOCUS	visual object detection with a computational attention system
WM	working memory
WTA	winner-takes-all

Bibliography

[AD07] G. Abdollahian and E. J. Delp. "Finding regions of interest in home videos based on camera motion". In: *Image Processing, 2007. ICIP 2007. IEEE International Conference on*. Vol. 4. 2007, pp. 545–548 (see p. 50).

[AKH93] Ketelaars-M. Adam J. J., H. Kingma, and T. Hoek. "On the time course and accuracy of spatial localization: basic data and a two-process model". In: *Acta Psychol.* 84.2 (1993), pp. 135–159 (see p. 7).

[ASC05] Z. Ambadar, J. W. Schooler, and J. F. Cohn. "Deciphering the enigmatic face: the importance of facial dynamics in interpreting subtle facial expressions." In: *Psychol. Sci.* 16.5 (2005), pp. 403–410 (see p. 36).

[Amd] *AMD Accelerated Parallel Processing OpenCL Programming Guide*. 2.4. AMD. 2012 (see p. 69).

[And98] J. Anderson. "Social stimuli and social rewards in primate learning and cognition". In: *Behav. Process.* 42.2-3 (1998), pp. 159–175 (see pp. 10, 14, 23).

[AC03] P. A. Arndt and H. Colonius. "Two stages in crossmodal saccadic integration: evidence from a visual-auditory focused attention task." In: *Exp. Brain Res.* 150.4 (2003), pp. 417–426 (see pp. 93, 129).

[BE94] W. F. Bacon and H. E. Egeth. "Overriding stimulus-driven attentional capture." In: *Percept. Psychophys.* 55.5 (1994), pp. 485–496 (see p. 6).

[BI11] F. Baluch and L. Itti. "Mechanisms of top-down attention." In: *Trends Neurosci.* 34.4 (2011), pp. 210–224 (see p. 5).

[BSL03] S-. W. Ban, J-. K. Shin, and M. Lee. "Face detection using biologically motivated saliency map model". In: *Proceedings of the International Joint Conference on Neural Networks*. Vol. 1. 2003, pp. 119–124 (see p. 50).

[BSA91] S. M. Banks, A. B. Sekuler, and S. J. Anderson. "Peripheral spatial vision: limits imposed by optics, photoreceptors, and receptor pooling". In: *J. Opt. Soc. Am. A* 8.11 (1991), pp. 1775–1787 (see p. 14).

[Bel+11] P. Belin, P. E. G. Bestelmeyer, M. Latinus, and R. Watson. "Understanding Voice Perception". In: *Brit. J. Psychol.* 102.4 (2011), pp. 711–725 (see pp. 93, 129).

[Ben+96] S. Bentin, T. Allison, A. Puce, E. Perez, and G. McCarthy. "Electrophysiological studies of face perception in humans". In: *J. Cognitive Neurosci.* 8.6 (1996), pp. 551–565 (see pp. 10, 13, 50).

[BTD00] L. E. Bernstein, P. E. Tucker, and M. E. Demorest. "Speech perception without hearing". English. In: *Percept. Psychophys.* 62 (2 2000), pp. 233–252 (see p. 36).

[BSB09] M. Bindemann, C. Scheepers, and A. M. Burton. "Viewpoint and center of gravity affect eye movements to human faces". In: *J. Vision* 9.2 (2009), pp. 7.1–16 (see p. 14).

[Bin+07] M. Bindemann, A. M. Burton, S. R. H. Langton, S. R. Schweinberger, and M. J. Doherty. "The control of attention to faces". In: *J. Vision* 7.10 (2007), pp. 15.1–8 (see p. 10).

[BBK08] E. Birmingham, W. F. Bischof, and A. Kingstone. "Gaze selection in complex social scenes". In: *Vis. Cogn.* 16.2 (2008), pp. 341–355 (see pp. 10, 23, 36).

[BBK09] E. Birmingham, W. F. Bischof, and A. Kingstone. "Saliency does not account for fixations to eyes within social scenes." In: *Vision Res.* 49.24 (2009), pp. 2992–3000 (see pp. 10, 36, 49).

[Bou70] H. Bouma. "Interaction effects in parafoveal letter recognition". In: *Nature* 226.5241 (1970), pp. 177–178 (see pp. 31, 35).

[BBS01] S. C. A. Braeutigam, A. J. Bailey, and S. J. Swithenby. "Task-dependent early latency (30-60 ms) visual processing of human faces and other objects." In: *NeuroReport* 12.7 (2001), pp. 1531–1536 (see pp. 10, 14).

[BS97] S. A. Brandt and L. W. Stark. "Spontaneous eye movements during visual imagery reflect the content of the visual scene". In: *J. Cognitive Neurosci.* 9.1 (1997), pp. 27–38 (see p. 7).

[BHF97] V. Brown, D. Huey, and J. M. Findlay. "Face detection in peripheral vision: do faces pop out?" In: *Perception* 26.12 (1997), pp. 1555–1570 (see p. 10).

[BP02] E. Bruno and D. Pellerin. "Robust motion estimation using spatial Gabor-like filters". In: *Signal Process.* 82 (2002), pp. 297–309 (see pp. 54, 80, 84).

[BPM07] J. N. Buchan, M. Paré, and K. G. Munhall. "Spatial statistics of gaze fixations during dynamic face processing." In: *Soc. Neurosci.* 2.1 (2007), pp. 1–13 (see p. 36).

[CI06] R. Carmi and L. Itti. "Visual causes versus correlates of attentional selection in dynamic scenes". In: *Vision Res.* 46.26 (2006), pp. 4333 –4345 (see p. 95).

[CW90] K. R. Cave and J. M. Wolfe. "Modeling the role of parallel processing in visual search." In: *Cognitive Psychol.* 22.2 (1990), pp. 225–271 (see p. 9).

[Cel] *Cell Broadband Engine Architecture*. 1.02. Sony. 2007 (see p. 69).

[CFK09] M. Cerf, E. P. Frady, and C. Koch. "Faces and text attract gaze independent of the task: Experimental data and computer model." In: *J. Vision* 9.12 (2009), pp. 10.1–15 (see pp. 9, 11).

[CFK08] M. Cerf, E. P. Frady, and C. Koch. "Using semantic content as cues for better scanpath prediction". In: *Proceedings of the 2008 symposium on Eye tracking research & applications*. Savannah, Georgia, 2008 (see p. 49).

[Cer+09] M. Cerf, J. Harel, A. Huth, W. Einhäuser, and C. Koch. "Decoding what people see from where they look: Predicting visual stimuli from scanpaths". In: *Attention in cognitive systems*. Ed. by Lucas Paletta and John K. Tsotsos. Berlin, Heidelberg: Springer-Verlag, 2009. Chap. Decoding What People See from Where They Look: Predicting Visual Stimuli from Scanpaths, pp. 15–26 (see p. 51).

[Cer+07] M. Cerf, J. Harel, W. Einhäuser, and C. Koch. "Predicting human gaze using low-level saliency combined with face detection." In: *NIPS'07*. 2007 (see pp. 9, 49, 50).

[Cha90] D. Chapman. "Instruction and Action". PhD thesis. Massachusetts Institute of Technology, 1990 (see p. 8).

[CL91] M. L. Cheal and D. R. Lyon. "Central and peripheral precuing of forced-choice discrimination". In: *Q. J. Exp. Psychol.* 43.4 (1991), pp. 859–880 (see p. 10).

[Che+03] L.-Q. Chen, X. Xie, X. Fan, W.-Y. Ma, H.-J. Zhang, and H.-Q. Zhou. "A visual attention model for adapting images on small displays". In: *Multimedia Syst.* 9.4 (2003), pp. 353–364 (see p. 49).

[CLL07] S. T. L. Chung, R. W. Li, and D. M. Levi. "Crowding between first- and second-order letter stimuli in normal foveal and peripheral vision." In: *J. Vision* 7.2 (2007), pp. 1–13 (see p. 14).

[CF89] J J Clark and N J Ferrier. "Control of visual attention in mobile robots". In: *Proceedings 1989 International Conference on Robotics and Automation*. 1989 (see p. 9).

[Com03] R. J. Compton. "The Interface Between Emotion and Attention: A Review of Evidence from Psychology and Neuroscience". In: *Behav. Cogn. Neurosci. Rev.* 2.2 (2003), pp. 115–129 (see p. 10).

[Cor+02] B. D. Corneil, M. Van Wanrooij, D. P. Munoz, and A. J. Van Opstal. "Auditory-Visual Interactions Subserving Goal-Directed Saccades in a Complex Scene". In: *J. Neurophysiol.* 88 (2002), pp. 438–454 (see pp. 93, 129).

[Cou+12] A. Coutrot, N. Guyader, G. Ionescu, and A. Caplier. "Influence of soundtrack on eye movements during video exploration." In: *J. Eye Mov. Res.* 5.4 (2012), pp. 1–10 (see pp. 93, 129).

[CKT10] S. M. Crouzet, H. Kirchner, and S. J. Thorpe. "Fast saccades toward faces: face detection in just 100 ms." In: *J. Vision* 10.4 (2010), pp. 16.1–17 (see pp. 10, 14).

[Cud] *CUDA C Programming Guide*. 5th ed. NVIDIA. 2012 (see p. 69).

[CT92] S M Culhane and J K Tsotsos. "A prototype for data-driven visual attention". In: *Proceedings 11th IAPR International Conference on Pattern Recognition*. Vol. 1. September. 1992, pp. 36–40 (see p. 9).

[CJT11] K. M. Curby, K. J. Johnson, and A. Tyson. "Face to face with emotion: Holistic face processing is modulated by emotional state." In: *Cognition Emotion* 26.1 (2011), pp. 1–10 (see p. 36).

[Cur+90] C. A. Curcio, K. R. Sloan, R. E. Kalina, and A. E. Hendrickson. "Human photoreceptor topography". In: *J. Comp. Neurol.* (1990), pp. 497–523 (see p. 14).

[DS96] H. Deubel and W. X. Schneider. "Saccade target selection and object recognition: evidence for a common attentional mechanism." In: *Vision Res.* 36.12 (1996), pp. 1827–1837 (see p. 7).

[DA95] D. Dong and J. Atick. "Statistics of natural time-varying images". In: *Network Computation in Neural Systems* 6.3 (1995), pp. 345–358 (see p. 36).

[Dor+10] M. Dorr, T. Martinetz, K. R. Gegenfurtner, and E. Barth. "Variability of eye movements when viewing dynamic natural scenes". In: *J. Vision* 10.10 (2010), pp. 1–17 (see pp. 58, 60, 62).

[Dow+81] B. M. Dow, A. Z. Snyder, R. G. Vautin, and R. Bauer. "Magnification factor and receptive field size in foveal striate cortex of the monkey." In: *Exp. Brain Res.* 44 (1981), pp. 213–228 (see p. 14).

[Dri+99] J. Driver, G. Davis, P. Ricciardelli, P. Kidd, E. Maxwell, and S. Baron-Cohen. "Gaze perception triggers reflexive visuospatial orienting". In: *Vis. Cogn.* 6.5 (1999), pp. 509–540 (see p. 9).

[Dua+11] L. Duan, C. Wu, H. Qiao, J. Gu, J. Miao, L. Qing, and Z. Yang. "Bio-inspired visual saliency detection and its application on image retargeting". In: *Proceedings of the 18th international conference on Neural Information Processing - Volume Part I*. ICONIP'11. Shanghai, China: Springer-Verlag, 2011, pp. 182–189 (see p. 51).

[Fan+12] Y. Fang, Z. Chen, W. Lin, and C.-W. Lin. "Saliency Detection in the Compressed Domain for Adaptive Image Retargeting". In: *IEEE Trans. Image Process.* 21.9 (2012), pp. 3888–3901 (see p. 9).

[FRW10] F. Farzin, S. M. Rivera, and D. Whitney. "Spatial resolution of conscious visual perception in infants." In: *Psychol. Sci.* 21.10 (2010), pp. 1502–1509 (see p. 14).

[Faw06] T. Fawcett. "An introduction to ROC analysis". In: *Pattern Recognit. Lett.* 27.8 (2006), pp. 861–874 (see p. 20).

[FW93] B. Fischer and H. Weber. In: *Behav. Brain Sci.* 16.3 (1993), pp. 553–567 (see p. 7).

[FW+08] S. Fletcher-Watson, J. M. Findlay, S. R. Leekam, and V. Benson. "Rapid detection of person information in a naturalistic scene". In: *Perception* 37.4 (2008), pp. 571–583 (see pp. 10, 26, 36).

[FRJ92] C. L. Folk, R. W. Remington, and J. C. Johnston. "Involuntary covert orienting is contingent on attentional control settings." In: *J. Exp. Psychol. Human* 18.4 (1992), pp. 1030–1044 (see p. 6).

[FRW94] C. L. Folk, R. W. Remington, and J. H. Wright. "The structure of attentional control: contingent attentional capture by apparent motion, abrupt onset, and color." In: *J. Exp. Psychol. Human* 20.2 (1994), pp. 317–329 (see p. 6).

[FFLM11] B. Follet, B. Fontaine, and O. Le Meur. "New insights into ambient and focal visual fixations using an automatic classification algorithm". In: *iPerception* 2.6 (2011), pp. 592–610 (see p. 36).

[Fox+00] E. Fox, V. Lester, R. Russo, R. J. Bowles, A. Pichler, and K. Dutton. "Facial Expressions of Emotion: Are Angry Faces Detected More Efficiently?" In: *Cognition Emotion* 14.1 (2000), pp. 61–92 (see p. 10).

[FG05] Russo R. Fox E. and G. A. Georgiou. "Anxiety modulates the degree of attentive resources required to process emotional faces". In: *Cogn. Affect. Behav. Ne.* 5.4 (2005), pp. 396–404 (see p. 10).

[GHV09] D. Gao, S. Han, and N. Vasconcelos. "Discriminant Saliency, the Detection of Suspicious Coincidences, and Applications to Visual Recognition". In: *IEEE T. Pattern. Anal. Mach. Intell.* 31.6 (2009), pp. 989–1005 (see pp. 9, 22).

[Geo+97] N. George, B. Jemel, N. Fiori, and B. Renault. "Face and shape repetition effects in humans: a spatio-temporal ERP study." In: *NeuroReport* 8.6 (1997), pp. 1417–1423 (see p. 10).

[GLR11] K. Guo, C. H. Liu, and H. Roebuck. "I know you are beautiful even without looking at you: discrimination of facial beauty in peripheral vision". In: *Perception* 40.2 (2011), pp. 191–195 (see p. 37).

[Guo+03] K. Guo, R. Robertson, S. Mahmoodi, Y. Tadmor, and M. Young. "How do monkeys view faces?—a study of eye movements". In: *Exp. Brain Res.* 150.3 (2003), pp. 363–374 (see p. 23).

[Guo+06] K. Guo, S. Mahmoodi, R. G. Robertson, and M. P. Young. "Longer fixation duration while viewing face images". In: *Exp. Brain Res.* 171.1 (2006), pp. 91–98 (see pp. 23, 26).

[GC11] F. F. E. Guraya and F. A. Cheikh. "Predictive visual saliency model for surveillance video". Anglais. In: *EUSIPCO proceedings*. 2011 (see p. 51).

[Gur+10] F. F. E. Guraya, F. A. Cheikh, A. Tremeau, Y. Tong, and H. Konik. "Predictive Saliency Maps for Surveillance Videos". In: *Proceedings of the 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science*. 2010 (see p. 51).

[HB11] H. Hadizadeh and I.V. Bajic. "Saliency-preserving video compression". In: *Multimedia and Expo (ICME), 2011 IEEE International Conference on*. 2011, pp. 1–6 (see p. 9).

[HZ07] B. Han and B. Zhou. "High Speed Visual Saliency Computation on GPU". In: *IEEE Int. Conf. Image Process.* Vol. 1. 2007, pp. 361–364 (see pp. 70, 103, 105).

[HKP07] J. Harel, C. Koch, and P. Perona. "Graph-based visual saliency". In: *Advances in Neural Information Processing Systems*. MIT Press, 2007, pp. 545–552 (see p. 50).

[Has+02] U. Hasson, I. Levy, M. Behrmann, T. Hendler, and R. Malach. "Eccentricity bias as an organizing principle for human high-order object areas." In: *Neuron* 34.3 (2002), pp. 479–490 (see p. 14).

[HS10] J. J. Heisz and D. I. Shore. "More efficient scanning for familiar faces." In: *J. Vision* 8.1 (2010), pp. 1–10 (see p. 14).

[Hen07] J. M. Henderson. "Regarding Scenes". In: *Current Directions in Psychological Science* 16.4 (2007), pp. 219–222 (see p. 23).

[HH99] J. M. Henderson and A. Hollingworth. "High-level scene perception". In: *Annu. Rev. Psychol.* 50 (1999), pp. 243–271 (see p. 22).

[HH05] O. Hershler and S. Hochstein. "At first sight: a high-level pop out effect for faces". In: *Vision Res.* 45 (2005), pp. 1707–1724 (see pp. 9, 10, 26).

[HH06] O. Hershler and S. Hochstein. "With a careful look: still no low-level confound to face pop-out". In: *Vision Res.* 46 (2006), pp. 3028–3035 (see pp. 10, 26).

[Her+10] O. Hershler, T. Golan, S. Bentin, and S. Hochstein. "The wide window of face detection." In: *J. Vision* 10.10 (2010), p. 21 (see pp. 14, 60).

[HJ01] H. Hill and A. Johnston. "Categorizing sex and identity from the biological motion of faces." In: *Curr. Biol.* 11.11 (2001), pp. 880–885 (see p. 36).

[Hil+10] S. Hillaire, G. Breton, N. Ouarti, R. Cozot, and A. Lécuyer. "Using a Visual Attention Model to Improve Gaze Tracking Systems in Interactive 3D Applications". In: *Computer Graphics Forum* 29.6 (2010), pp. 1830–1841 (see p. 70).

[HSP12] P. J. Hills, A. J. Sullivan, and J. M. Pake. "Aberrant first fixations when looking at inverted faces in various poses: The result of the centre-of-gravity effect?" In: *Brit. J. Psychol.* 103.4 (2012), pp. 520–538 (see p. 36).

[Ho+03] C-.C. Ho, W-. H. Cheng, T-. J. Pan, and J-. L. Wu. "A user-attention based focus detection framework and its applications". In: *Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on*. Vol. 3. 2003, pp. 1315 –1319 (see p. 50).

[HPGGD10] T. Ho-Phuoc, N. Guyader, and A. Guérin-Dugué. "A Functional and Statistical Bottom-Up Saliency Model to Reveal the Relative Contributions of Low-Level Visual Guiding Factors". In: *Cogn. Comput.* 2.4 (2010), pp. 344–359 (see p. 49).

[Hof98] J. E. Hoffman. "Visual attention and eye movements". In: *Attention*. Ed. by HEditor Pashler. Vol. 31. 1992. Psychology Press, 1998, pp. 119–153 (see p. 7).

[HS95] J. E. Hoffman and B. Subramaniam. "The role of visual attention in saccadic eye movements". In: *Percept. Psychophys.* 57 (1995), pp. 787–795 (see p. 7).

[HHR10] D. Houzet, S. Huet, and A. Rahman. "SysCellC: a data-flow programming model on multi-GPU". In: *Procedia Computer Science*. Vol. 1. 1. May 2010, pp. 1029–1038.

[HC08] J. H.-W Hsiao and G. Cottrell. "Two fixations suffice in face recognition." In: *Psychol. Sci.* 19.10 (2008), pp. 998–1006 (see p. 36).

[Itt05] L. Itti. "Quantifying the contribution of low-level saliency to human eye movements in dynamic scenes". In: *Vis. Cogn.* 12.6 (2005), pp. 1093–1123 (see p. 69).

[IKN98] L. Itti, C. Koch, and E. Niebur. "A Model of Saliency-Based Visual Attention for Rapid Scene Analysis". In: *IEEE T. Pattern. Anal. Mach. Intell.* 20 (1998), pp. 1254–1259 (see pp. 9, 19, 49, 50, 53).

[JR04] C. Jacques and B. Rossion. "Concurrent processing reveals competition between visual representations of faces." In: *Neuroreport* 15.15 (2004), pp. 2417–2421 (see p. 14).

[JR06] C. Jacques and B. Rossion. "The time course of visual competition to the presentation of centrally fixated faces." In: *J. Vision* 6.2 (2006), pp. 154–162 (see p. 14).

[Jeb+09] N. Jebara, D. Pins, P. Despretz, and M. Boucart. "Face or building superiority in peripheral vision reversed by task requirements". In: *Advances in Cognitive Psychology* 5 (2009), pp. 42–53 (see p. 14).

[Jef96] D. A. Jeffreys. "Evoked Potential Studies of Face and Object Processing". In: *Vis. Cogn.* 3.1 (1996), pp. 1–38 (see pp. 93, 129).

[JQ10] P. Jiang and X. Qin. "Keyframe-Based Video Summary Using Visual Attention Clues". In: *IEEE Multimedia* 17.2 (2010), pp. 64 –73 (see pp. 104, 105).

[JG10] A. Johnson and R. Gurnsey. "Size scaling compensates for sensitivity loss produced by a simulated central scotoma in a shape-from-texture task." In: *J. Vision* 10.12 (2010), pp. 1–16 (see p. 14).

[JM01] M. H. Johnson and D. Mareschal. "Cognitive and perceptual development during infancy." In: *Curr. Opin. Neurobiol.* 11.2 (2001), pp. 213–218 (see pp. 10, 26).

[JHC92] A. Johnston, H. Hill, and N. Carman. "Recognising faces: effects of lighting direction, inversion, and brightness reversal." In: *Perception* 21.3 (1992), pp. 365–375 (see p. 36).

[Jos+05] T. Jost, N. Ouerhani, R. Von Wartburg, R. Muri, and H. Hugli. "Assessing the contribution of color in visual attention". In: *Computer Vision and Image Understanding* 100.1-2 (2005), pp. 107–123 (see p. 22).

[Jud+09] T. Judd, K. Ehinger, F. Durand, and A. Torralba. "Learning to predict where humans look". In: *Computer Vision, 2009 IEEE 12th International Conference on*. 2009, pp. 2106 –2113 (see p. 58).

[JC76] M. A. Just and P. A. Carpenter. "Eye fixations and cognitive processes". In: *Cognitive Psychol.* 8.4 (1976), pp. 441–480 (see p. 23).

[Kan+09] C. Kanan, M. H. Tong, L. Zhang, and G. W. Cottrell. "SUN: Top-down saliency using natural statistics". In: *Vis. Cogn.* 17.6-7 (2009), pp. 979–1003 (see p. 9).

[KY06] N. Kanwisher and G. Yovel. "The fusiform face area: a cortical region specialized for the perception of faces". In: *Philos. Trans. R. Soc. London, Ser. B* 361.1476 (2006), pp. 2109–2128 (see pp. 10, 13, 26, 50).

[KU01] S. Kastner and L. G. Ungerleider. "The neural basis of biased competition in human visual cortex." In: *Neuropsychologia* 39.12 (2001), pp. 1263–1276 (see p. 14).

[Kna03] B. Knappmeyer. "The use of facial motion and facial form during the processing of identity". In: *Vision Res.* 43.18 (2003), pp. 1921–1936 (see p. 36).

[KU85] C. Koch and S. Ullman. "Shifts in selective visual attention: towards the underlying neural circuitry". In: *Hum. Neurobiol.* 4 (1985), pp. 219–227 (see pp. 8, 49).

[Kuc11] J. Kucerova. "Saliency map augmentation with facial detection". In: *CESCG*. 2011 (see p. 50).

[LB99] S. Langton and V. Bruce. "Reflexive Visual Orienting in Response to the Social Attention of Others". In: *Vis. Cogn.* 6.5 (1999), pp. 541–567 (see p. 9).

[LM03] C. R. Lansing and G. W. McConkie. "Word identification and eye fixation locations in visual and visual-plus-auditory presentations of spoken sentences." In: *Percept. Psychophys.* 65.4 (2003), pp. 536–552 (see p. 36).

[LRR03] N. Lavie, T. Ro, and C. Russell. "The role of perceptual load in processing distractor faces". In: *Psychol. Sci.* 14.5 (2003), pp. 510–515 (see p. 36).

[LMLCB07] O. Le Meur, P. Le Callet, and D. Barba. "Predicting visual fixations on video based on low-level visual features". In: *Vision Res.* 47.19 (2007), pp. 2483–2498 (see pp. 49, 58).

[LM+06] O. Le Meur, P. Le Callet, D. Barba, and D. Thoreau. "A coherent computational approach to model bottom-up visual attention". In: *IEEE T. Pattern. Anal. Mach. Intell.* 28.5 (2006), pp. 802–817 (see pp. 19, 22).

[LM+10] O. Le Meur, A. Ninassi, P. Le Callet, and D. Barba. "Overt visual attention for free-viewing and quality assessment tasks". In: *Signal Process. Image Commun.* 25.7 (Aug. 2010), pp. 547–558 (see p. 20).

[LKA85] D. M. Levi, S. A. Klein, and A. P. Aitsebaomo. "Vernier acuity, crowding and cortical magnification". In: *Vision Res.* 25.7 (1985), pp. 963–977 (see p. 14).

[Lev+01] I. Levy, U. Hasson, G. Avidan, T. Hendler, and R. Malach. "Center-periphery organization of human object areas." In: *Nat. Neurosci.* 4.5 (2001), pp. 533–539 (see p. 14).

[LN08] H. Li and K. N. Ngan. "Saliency model-based face segmentation and tracking in head-and-shoulder video sequences". In: *J. Visual Commun. Image Represent.* 19.5 (2008), pp. 320 –333 (see p. 51).

[Liu+09] H. Liu, Y. Agam, J. R. Madsen, and G. Kreiman. "Timing, timing, timing: fast decoding of object information from intracranial field potentials in human visual cortex." In: *Neuron* 62.2 (2009), pp. 281–290 (see pp. 10, 13).

[LHK02] J. Liu, A. Harris, and N. Kanwisher. "Stages of processing in face perception: an MEG study". In: *Nat. Neurosci.* 5.9 (2002), pp. 910–916 (see p. 10).

[Liu+10] Z. Liu, H. Yan, L. Shen, K. N. Ngan, and Z. Zhang. "Adaptive image retargeting using saliency-based continuous seam carving". In: *Opt. Eng.* 49.1 (2010), pp. 1–10 (see p. 9).

[Lu+10] T. Lu, Z. Yuan, Y. Huang, D. Wu, and H. Yu. "Video retargeting with nonlinear spatial-temporal saliency". In: *IEEE Int. Conf. on Image Process.* 2010 (see pp. 104, 105).

[MZ08] Q. Ma and L. Zhang. "Saliency-Based Image Quality Assessment Criterion". In: *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues*. Ed. by De-Shuang Huang, Donald Wunsch, Daniel Levine, and Kang-Hyun Jo. Vol. 5226. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2008, pp. 1124–1133 (see p. 9).

[Ma+05] Y. F. Ma, X. S. Hua, L. Lu, and H. J. Zhang. "A generic framework of user attention model and its application in video summarization". In: *IEEE T. Multimedia.* 7 (2005), pp. 907–919 (see p. 49).

[MLH02] R. Malach, I. Levy, and U. Hasson. "The topography of high-order human object areas." In: *Trends Cogn. Sci.* 6.4 (2002), pp. 176–184 (see p. 14).

[Mar10] S. Marat. "Modèles de saillance visuelle par fusion d'informations sur la luminance, le mouvement et les visages pour la prédiction de mouvements oculaires lors de l'exploration de vidéos." PhD thesis. Université Joseph-Fourier - Grenoble I, 2010 (see pp. 3, 11, 124).

[MGP09] S. Marat, N. Guyader, and D. Pellerin. "Recent advances in signal processing". In: ed. by Ashraf A. Zaher. 12. In-Tech, 2009. Chap. Gaze prediction improvement by adding a face feature to a saliency model, pp. 195–210 (see p. 50).

[Mar+13] S. Marat, A. Rahman, D. Pellerin, N. Guyader, and D. Houzet. "Improving visual saliency by adding 'face feature map' and 'center bias'". In: *Cogn. Comput.* 5.1 (2013), pp. 63–75.

[Mar+09] S. Marat, T. H. Phuoc, L. Granjon, N. Guyader, D. Pellerin, and A. Guérin-Dugué. "Modelling Spatio-Temporal Saliency to Predict Gaze Direction for Short Videos". In: *Int. J. Comput. Vision* 82 (2009), pp. 231–243 (see pp. 3, 11, 15, 49, 51, 57, 61, 92, 95, 104, 105, 124, 126).

[McC+88] G. W. McConkie, P. W. Kerr, M. D. Reddix, and D. Zola. "Eye movement control during reading: I. The location of initial eye fixations on words". In: *Vision Res.* 28.10 (1988), pp. 1107–1118 (see p. 36).

[MMN99] R. M. McPeek, V. Maljkovic, and K. Nakayama. "Saccades require focal attention and are facilitated by a short-term memory system." In: *Vision Res.* 39.8 (1999), pp. 1555–1566 (see p. 7).

[Mec+04] A. Mechelli, C. J. Price, K. J. Friston, and A. Ishai. "Where bottom-up meets top-down: neuronal interactions during perception and imagery." In: *Cereb. Cortex* 14.11 (2004), pp. 1256–65 (see pp. 10, 26, 50).

[Mel+00] D. R. Melmoth, H. T. Kukkonen, P. K. Mäkelä, and J. M. Rovamo. "The effect of contrast and size scaling on face perception in foveal and extrafoveal vision." In: *Investigative Ophthalmology & Visual Science* 41.3 (2000), pp. 948–954 (see p. 14).

[MGP95] R. Milanese, S. Gil, and T. Pun. "Attentive mechanisms for dynamic and static scene analysis". In: *Opt. Eng.* 34.8 (1995), pp. 2428–2434 (see p. 9).

[MGG93] E. K. Miller, P. M. Gochin, and C. G. Gross. "Suppression of visual responses of neurons in inferior temporal cortex of the awake macaque by addition of a second stimulus." In: *Brain Res.* 616.1-2 (1993), pp. 25–29 (see p. 14).

[MG06] A.D. Milner and M.A. Goodale. *The visual brain in action*. Oxford psychology series. Oxford University Press, 2006 (see p. 55).

[Mit+10] P. K. Mital, T. J. Smith, R. L. Hill, and J. M. Henderson. "Clustering of Gaze During Dynamic Scene Viewing is Predicted by Motion". In: *Cogn. Comput.* 3.1 (2010), pp. 5–24 (see pp. 49, 61).

[MB99] K. Mogg and B. P. Bradley. "Orienting of attention to threatening facial expressions presented under conditions of restricted awareness." In: *Cognition Emotion* 13.6 (1999), pp. 713–740 (see p. 10).

[MF88] H. J. Müller and J. M. Findlay. "The effect of visual attention on peripheral discrimination thresholds in single and multiple element displays." In: *Acta Psychol.* 69.2 (1988), pp. 129–155 (see p. 10).

[MR89] H. J. Müller and P. M. Rabbitt. "Reflexive and voluntary orienting of visual attention: time course of activation and resistance to interruption." In: *J. Exp. Psychol. Human* 15.2 (1989), pp. 315–330 (see p. 10).

[Mun] Aaftab Munshi. *The OpenCL Specification*. 1.2. Khronos OpenCL Working Group (see p. 69).

[NS71] D. Noton and L. Stark. "Scanpaths in eye movements during pattern perception." In: *Science* 171.968 (1971), pp. 308–311 (see p. 7).

[Kep] *NVIDIA's Next Generation CUDA Compute Architecture: Kepler GK110* (see p. 114).

[OB95] J. M. Odobez and P. Bouthemy. "Robust Multiresolution Estimation of Parametric Motion Models Applied to Complex Scenes". In: *J. Visual Commun. Image Represent.* 6 (1995), pp. 348–365 (see p. 54).

[OLE01] A. Öhman, D. Lundqvist, and F. Esteves. "The face in the crowd revisited: a threat advantage with schematic stimuli". In: *J. Pers. Soc. Psychol.* 80.3 (2001), pp. 381–96 (see p. 10).

[OAVE93] B. A. Olshausen, C. E. Anderson, and D. C. Van Essen. "A neural model of visual attention and invariant pattern recognition". In: *J. Neurosci.* 13 (1993), pp. 4700–4719 (see p. 9).

[OP92] M. W. Oram and D. I. Perrett. "Time course of neural responses discriminating different views of the face and head." In: *J. Neurophysiol.* 68.1 (1992), pp. 70–84 (see p. 10).

[OH03] N. Ouerhani and H. Hügli. "Real-time visual attention on a massively parallel SIMD architecture". In: *Real-Time Imaging* 9 (2003), pp. 189–196 (see p. 69).

[Pal99] S.E. Palmer. *Vision Science: Photons to Phenomenology*. Bradford Books. MIT Press, 1999 (see p. 5).

[Par+03] C. L. Paras, J. A. Yamashita, M. L. Simas, and M. A. Webster. "Face perception and configural uncertainty in peripheral vision". In: *J. Vision* 3.9 (2003), p. 822 (see p. 14).

[PLN02] D. Parkhurst, K. Law, and E. Niebur. "Modeling the role of salience in the allocation of overt visaul attention". In: *Vision Res.* 42 (2002), pp. 107–123 (see pp. 6, 19).

[PN04] D. J. Parkhurst and E. Niebur. "Texture contrast attracts overt visual attention in natural scenes." In: *Eur. J. Neurosci.* 19.3 (2004), pp. 783–789 (see p. 6).

[PPM04] D. G. Pelli, M. Palomares, and N. J. Majaj. "Crowding is unlike ordinary masking: distinguishing feature integration from detection." In: *J. Vision* 4.12 (2004), pp. 1136–1169 (see p. 14).

[PI08] R. J. Peters and L. Itti. "Applying computational tools to predict gaze direction in interactive visual environments". In: *ACM T. Appl. Percept.* 5.2 (2 2008), pp. 1–9 (see p. 49).

[Pet+05] R. J. Peters, A. Iyer, L. Itti, and Koch C. "Components of bottom-up gaze allocation in natural images". In: *Vision Res.* 45 (2005), pp. 2397–2416 (see pp. 19–21).

[Phi+00] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss. "The FERET evaluation methodology for face-recognition algorithms". In: *IEEE T. Pattern. Anal. Mach. Intell.* 22.10 (2000), pp. 1090–1104 (see pp. 93, 129).

[Pic+07] A. Picot, G. Bailly, F. Elisei, and S. Raidt. "Scrutinizing natural scenes: controlling the gaze of an embodied conversational agent". In: *Proceedings of the 7th international conference on Intelligent Virtual Agents*. IVA '07. Paris, France: Springer-Verlag, 2007, pp. 272–282 (see p. 51).

[PS10] N. Pitsianis and X. Sun. "Fast Extraction of Feature Salience Maps for Rapid Video Data Analysis". In: *Proceedings of the 2010 High Performance Embedded Computing (HPEC)*. 2010 (see p. 70).

[PS00] C. M. Privitera and L. W. Stark. "Algorithms for defining visual regions-of-interest: comparison with eye fixations". In: *IEEE T. Pattern. Anal. Mach. Intell.* 22.9 (2000), pp. 970–982 (see p. 9).

[Dir] *Programming Guide for Direct3D*. 11.1. Microsoft. 2012 (see p. 69).

[QXG09] F. Qi, Song X., and Shi G. "LDA based color information fusion for visual objects tracking". In: *IEEE Int. Conf. on Image Process.* 2009, pp. 2201 –2204 (see p. 105).

[RHP11] A. Rahman, D. Houzet, and D. Pellerin. "Visual Saliency Model on Multi-GPU". In: *GPU Computing Gems Emerald Edition*. Elsevier, 2011, pp. 451–472.

[RPH12] A. Rahman, D. Pellerin, and D. Houzet. "Face perception: Influence of location and number in videos". In: *Image Analysis for Multimedia Interactive Services (WIAMIS), 2012 13th International Workshop on*. Dublin, Ireland, 2012, pp. 1–4 (see p. 37).

[Rah+10] A. Rahman, D. Houzet, D. Pellerin, and L. Agud. "GPU implementation of motion estimation for visual saliency". In: *Proceedings of the Conference on Design and Architectures for Signal and Image Processing (DASIP 2010)*. Edinburgh, United Kingdom, Oct. 2010, pp. 222–227.

[Rah+11a] A. Rahman, D. Houzet, D. Pellerin, S. Marat, and N. Guyader. "Parallel implementation of a spatio-temporal visual saliency model". In: *Journal of Real-Time Image Processing* 6.1 (2011), pp. 3–14.

[Rah+11b] A. Rahman, G. Song, D. Pellerin, and D. Houzet. "Spatio-temporal fusion of visual attention model". In: *Proc. European Signal Processing Conference (EUSIPCO)*. Barcelona, Spain, 2011, pp. 2029–2033.

[RBC06] U. Rajashekar, A. C. Bovik, and L. K. Cormack. "Visual search in noise: revealing the influence of structural cues by gaze-contingent classification image analysis." In: *J. Vision* 6.4 (2006), pp. 379–386 (see p. 22).

[Raj+08] U. Rajashekar, I. Van Der Linde, A. C. Bovik, and L. K. Cormack. "GAFFE: A Gaze-Attentive Fixation Finding Engine". In: *IEEE Trans. Image Process.* 17.4 (2008), pp. 564–573 (see pp. 22, 50).

[RAK09] K. Rapantzikos, Y. Avrithis, and S. Kollias. "Dense saliency-based spatiotemporal feature points for action recognition". In: *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. 2009, pp. 1454 –1461 (see p. 9).

[Ray98] K. Rayner. "Eye movements in reading and information processing: 20 years of research." In: *Psychol. Bull.* 124.3 (1998), pp. 372–422 (see p. 23).

[RRK06] L. Reddy, L. Reddy, and C. Koch. "Face identification in the near-absence of focal attention." In: *Vision Res.* 46.15 (2006), pp. 2336–2343 (see p. 14).

[RVC07] L. W. Renninger, P. Verghese, and J. Coughlan. "Where to look next? Eye movements reduce local uncertainty". In: *J. Vision* 7 (2007), pp. 1–17 (see p. 58).

[Rig+11] S. Rigoulot, F. D'Hondt, S. Defoort-Dhellemmes, P. Despretz, J. Honoré, and H. Sequeira. "Fearful faces impact in peripheral vision: behavioral and neural evidence." In: *Neuropsychologia* 49.7 (2011), pp. 2013–2021 (see p. 14).

[RRL01] T. Ro, C. Russell, and N. Lavie. "Changing faces: A detection advantage in the flicker paradigm". In: *Psychol. Sci.* 12.1 (2001), pp. 94–99 (see pp. 10, 14).

[RT95] E. T. Rolls and M. J. Tovee. "The responses of single neurons in the temporal visual cortical areas of the macaque when more than one stimulus is present in the receptive field". In: *Exp. Brain Res.* 103.3 (1995), pp. 409–420 (see p. 14).

[Ros+00] B. Rossion, I. Gauthier, M. J. Tarr, P. Despland, R. Bruyer, S. Linotte, and M. Crommelinck. "The N170 occipito-temporal component is delayed and enhanced to inverted faces but not to inverted objects: an electrophysiological account of face-specific processes in the human brain". In: *Neuroreport* 11.1 (2000), pp. 69–74 (see pp. 10, 14, 23).

[RMFT03] G A Rousselet, M J M Macé, and M Fabre-Thorpe. "Is it an animal? Is it a human face? Fast processing in upright and inverted natural scenes." In: *J. Vision* 3.6 (2003), pp. 440–55 (see p. 14).

[Rou+05] G. A. Rousselet, J. S. Husk, P. J. Bennett, and A. B. Sekuler. "200 ms of controversies: A high-density ERP study of face processing". In: *J. Vision* 5.8 (2005), p. 819 (see p. 14).

[Rut+04] U. Rutishauser, D. Walther, C. Koch, and P. Perona. "Is bottom-up attention useful for object recognition?" In: *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*. Vol. 2. 2004, pp. 37–44 (see p. 9).

[SS12] B. Schauerte and R. Stiefelhagen. "Predicting human gaze using quaternion DCT image signature saliency and face detection." In: *WACV*. IEEE, 2012, pp. 137–144 (see pp. 9, 11, 50).

[SM10] H. J. Seo and P. Milanfar. "Visual saliency for automatic target detection, boundary detection, and image quality assessment". In: *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. 2010, pp. 5578–5581 (see p. 9).

[SCH09] P. Sharma, F. A. Cheikh, and J. Y. Hardeberg. "Face saliency in various human visual saliency models". In: *Image and Signal Processing and Analysis, 2009. ISPA 2009. Proceedings of 6th International Symposium on*. 2009, pp. 327 –332 (see pp. 9, 11, 50).

[SCH08] P. Sharma, F. A. Cheikh, and J. Y. Hardeberg. "Saliency Map for Human Gaze Prediction in Images". In: *Sixteenth Color Imaging Conference: Color Science and Engineering Systems, Technologies, and Applications*. 2008 (see p. 50).

[SFH86] M. Shepherd, J. M. Findlay, and R. J. Hockey. "The relationship between eye movements and spatial attention." In: *Q. J. Exp. Psychol.* 38.3 (1986), pp. 475–491 (see p. 7).

[SM89] M. Shepherd and H. J. Müller. "Movement versus focusing of visual attention." In: *Percept. Psychophys.* 46.2 (1989), pp. 146–154 (see p. 10).

[SPG12] G. Song, D. Pellerin, and L. Granjon. "How different kinds of sound in videos can influence gaze?" In: *Image Analysis for Multimedia Interactive Services (WIAMIS), 2012 13th International Workshop on*. Dublin, Ireland, 2012, pp. 1–4 (see pp. 93, 95, 129).

[SPG11] G. Song, D. Pellerin, and L. Granjon. "Sound effect on visual gaze when looking at videos". In: *19th European Signal Processing Conference (EUSIPCO 2011)*. Barcelona, Spain, 2011, pp. 2034–2038 (see p. 95).

[STB89] D. L. Still, L. N. Thibos, and A. Bradley. "Peripheral image quality is almost as good as central image quality." In: *Invest. Ophth. Vis. Sci.* 30 (1989), p. 52 (see p. 14).

[SMS12] Y. Sugano, Y. Matsushita, and Y. Sato. "Appearance-based gaze estimation using visual saliency". In: *IEEE T. Pattern. Anal. Mach. Intell.* PP.99 (2012), p. 1 (see p. 50).

[Sum+06] C. Summerfield, T. Egner, M. Greene, E. Koechlin, J. Mangels, and J. Hirsch. "Predictive codes for forthcoming perception in the frontal cortex." In: *Science* 314.5803 (2006), pp. 1311–4 (see pp. 10, 26, 50).

[Tat07] B. W. Tatler. "The central fixation bias in scene viewing: selecting an optimal viewing position independently of motor biases and image feature distributions." In: *J. Vision* 7.14 (2007), pp. 4.1–17 (see pp. 58, 65).

[TBG05] B.W. Tatler, R. J. Baddeley, and I. D. Gilchrist. "Visual correlates of fixation selection: effects of scale and time". In: *Vision Res.* 45 (2005), pp. 643–659 (see pp. 19, 20).

[Tau09] J. Taubert. "Chimpanzee faces are "special" to humans." In: *Perception* 38.3 (2009), pp. 343–356 (see pp. 10, 26).

[The90] J. Theeuwes. "Perceptual selectivity is task dependent: evidence from selective search." In: *Acta Psychol.* 74.1 (1990), pp. 81–99 (see p. 6).

[The94] J. Theeuwes. "Stimulus-driven capture and attentional set: selective search for color and visual abrupt onsets." In: *J. Exp. Psychol. Human* 20.4 (1994), pp. 799–806 (see p. 6).

[Tho+01] S. J. Thorpe, K. R. Gegenfurtner, M. Fabre-Thorpe, and H. H. Bülthoff. "Detection of animals in natural images using far peripheral vision." In: *Eur. J. Neurosci.* 14.5 (2001), pp. 869–876 (see p. 37).

[TJC09] P. Tomalski, M. H. Johnson, and G. Csibra. "Temporal-nasal asymmetry of rapid orienting to face-like stimuli". In: *NeuroReport* 20.15 (2009), pp. 1309–1312 (see p. 37).

[Ton+10] Y. Tong, H. Konik, F. A. Cheikh, and A. Trémeau. "Full Reference Image Quality Assessment Based on Saliency Map Analysis". Anglais. In: *J. Imaging Sci. Technol.* 54.3 (June 2010). ISBN / ISSN: 1062-3701, pp. 1–14 (see pp. 9, 11, 50).

[Tor+06] A. Torralba, A. Oliva, M. S. Castelhano, and J. M. Henderson. "Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search." In: *Psychol. Rev.* 113.4 (2006), pp. 766–786 (see pp. 9, 19–21).

[TG80] A. M. Treisman and G. Gelade. "A feature-integration theory of attention". In: *Cognitive Psychol.* 12 (1980), pp. 97–136 (see p. 8).

[TRP07] N. Tsapatsoulis, K. Rapantzikos, and C. Pattichis. "An embedded saliency map estimator scheme: application to video encoding." In: *Int. J. Neural Syst.* 17.4 (2007), pp. 289–304 (see p. 50).

[Tse+09] P. Tseng, R. Carmi, I. G. M. Cameron, D.P. Munoz, and L. Itti. "Quantifying center bias of observers in free viewing of dynamic natural scenes". In: *J. Vision* 9.7 (2009), pp. 1–16 (see pp. 58, 65).

[Tso+95] J. K. Tsotsos, S. M. Culhane, W. Y. K. Wai, Y. Lai, N. Davis, and F. Nuflo. "Modeling Visual Attention via Selective Tuning". In: *Artif. Intell.* 78 (1995), pp. 507–545 (see p. 49).

[Vo+12] M. L.-H. Võ, T. J. Smith, P. K. Mital, and J. M. Henderson. "Do the eyes really have it? Dynamic allocation of attention when viewing moving faces". In: *J. Vision* 12.13 (2012), pp. 1–14 (see pp. 36, 93, 129).

[VR+98] R. Van Rullen, J. Gautrais, A. Delorme, and S. Thorpe. "Face processing using one spike per neurone." In: *Bio Systems* 48.1-3 (1998), pp. 229–239 (see pp. 93, 129).

[Vel02] B. M. Velichkovsky. "Heterarchy of cognition: the depths and the highs of a framework for memory research." In: *Memory Hove England* 10.5-6 (2002), pp. 405–19 (see p. 36).

[VJ04] P. Viola and M. J. Jones. "Robust Real-Time Face Detection". In: *Int. J. Comput. Vision* 57 (2 2004), pp. 137–154 (see pp. 49–51, 100).

[VR79] V. Virsu and J. Rovamo. "Visual resolution, contrast sensitivity, and the cortical magnification factor." In: *Exp. Brain Res.* 37.3 (1979), pp. 475–494 (see p. 14).

[Viv90] P. Viviani. "Eye movements and their role in visual and cognitive processes." In: ed. by E. Kowler. Elsevier Science, 1990. Chap. 8, Viviani, P. (See p. 7).

[Vui00] P. Vuilleumier. "Faces call for attention: evidence from patients with visual extinction". In: *Neuropsychologia* 38.5 (2000), pp. 693 –700 (see pp. 10, 14).

[WWZ10] C. Wang, Y. Wang, and Z. Zhang. "Face Detection in Videos Using Skin Color Segmentation and Saliency Model". In: *Pattern Recognition (CCPR), 2010 Chinese Conference on*. 2010, pp. 1 –5 (see pp. 9, 11, 50).

[WP12] H.-C. Wang and M. Pomplun. "The attraction of visual attention to texts in real-world scenes". In: *J. Vision* 12.6 (2012), pp. 1–17 (see p. 19).

[WB02] Z. Wang and A. C. Bovik. "A universal image quality index". In: *IEEE Signal. Proc. Let.* 9 (2002), pp. 81–84 (see p. 84).

[WG97] J. Wolfe and G. Gancarz. "Guided Search 3.0: A model of visual search catches up with Jay Enoch 40 years later". In: *Basic and Clinical Applications of Vision Science*. Ed. by V Lakshminarayanan. Norwell, MA: Kluwer Academic Publishers, 1997, pp. 189–192 (see p. 8).

[WH04] J. M. Wolfe and T. S. Horowitz. "What attributes guide the deployment of visual attention and how do they do it?" In: *Nat. Rev. Neurosci.* 5 (2004), pp. 1–7 (see p. 61).

[XXR10] X. Xiao, C. Xu, and Y. Rui. "Video based 3D reconstruction using spatio-temporal attention analysis". In: *Proc. IEEE Int. Conf. on Multimedia and Expo*. 2010, pp. 1091–1096 (see p. 105).

[Xu+09] T. Xu, T. Pototschnig, K. Kuhnlenz, and M. Buss. "A high-speed multi-GPU implementation of bottom-up attention using CUDA". In: *Proceedings of the IEEE International Conference on Robotics and Automation*. Ieee, 2009, pp. 41–47 (see p. 70).

[YJ96] S. Yantis and J. Jonides. "Attentional capture by abrupt onsets: new perceptual objects or visual masking?" In: *J. Exp. Psychol. Human* 22.6 (1996), pp. 1505–1513 (see p. 6).

[Yan+11] V. Yanulevskaya, J. B. Marsman, F. Cornelissen, and J.-M. Geusebroek. "An Image Statistics-Based Model for Fixation Prediction". In: *Cogn. Comput.* 3.1 (2011), pp. 94–104 (see p. 49).

[Yar67] A. L. Yarbus. *Eye movements and vision*. Ed. by EnglishEditor Trans By L A Riggs. Vol. chapter VI. Plenum Press, 1967, p. 222 (see pp. 7, 23).

[Yin69] R. Yin. "Looking at upside down faces". In: *J. of Exp. Psychol.* 81 (1969), pp. 141–145 (see p. 10).

[YV95] I. T. Young and L. J. van Vliet. "Recursive implementation of the Gaussian filter". In: *Signal Process.* 44.2 (1995), pp. 139–151 (see p. 78).

[Zha+08] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell. "SUN: a bayesian framework for saliency using natural statistics". In: *J. Vision* 8.7 (2008), pp. 1–20 (see p. 58).

[ZK11] Q. Zhao and C. Koch. "Learning a saliency map using fixated locations in natural scenes". In: *J. Vision* 11.3 (2011), pp. 1–15 (see p. 58).